

NASA Technical Memorandum 78681

COMPUTER PROGRAM FOR DETERMINING MASS PROPERTIES OF A RIGID STRUCTURE

(NASA-TM-78681) COMPUTER PROGRAM FOR
DETERMINING MASS PROPERTIES OF A RIGID
STRUCTURE (NASA) 47 F HC A03/MF A01

N78-20535

CSCI 20K

Unclas
11902

G3/39

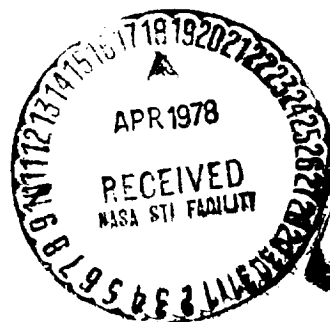
REID A. HULL, JOHN L. GILBERT AND
PHILLIP J. KLICH

MARCH 1978



National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665



NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

COMPUTER PROGRAM FOR DETERMINING MASS PROPERTIES OF A

RIGID STRUCTURE

BY Reid A. Hull, Dr. John L. Gilbert and Phillip J. Klich

SUMMARY

A computer program (ND0702) has been developed for the rapid computation of the mass properties of complex structural systems. The program uses rigid body analyses and permits differences in structural material throughout the total system. It is based on the premise that complex systems can be adequately described by a combination of basic elemental shapes. The following thirteen widely used structural shapes were selected for inclusion in the program:

1. Discrete mass
2. Cylinder
3. Truncated cone
4. Torus
5. Beam (arbitrary cross section)
6. Circular rod (arbitrary cross section)
7. Spherical segment
8. Sphere
9. Hemisphere
10. Parallelepiped
11. Swept Trapezoidal Panel
12. Symmetric Trapezoidal Panels
13. Curved Rectangular Panel

Simple geometric data describing size and location of each element and the respective material density or weight of each element are the only required input data. From this minimum input, the program yields system weight, center of gravity, moments of inertia and products of inertia with respect to mutually perpendicular axes through the system center of gravity. The program also yields mass properties of the individual shapes relative to component axes.

Permanent configuration records and the use of iterative calculations to investigate design systems or to determine optimums contribute to the cost-effectiveness of the programs use.

INTRODUCTION

Determining the mass properties of any rigid structure is a problem that at times becomes complex, but one which can easily be dealt with utilizing computer solutions.

For rigid structures the solution of the mass properties requires transformation to an axis parallel to the system axis and becomes laborious almost to the point of being impractical.

Any complex structure must be broken down into elements in order to exact a solution. The approach selected for the program presented in this paper was to automate the input to the point where an element's shape, geometry, density or weight, and three grid points are the only requirements. This approach was influenced by the simplicity of computing the direction cosines (Euler angle relationship) from the given three grid points. The program as outlined in this paper performs essentially the same process as calculations "by hand" and is extremely useful for rigid structures skewed in space. This program also provides improved accuracy, time savings, and complete permanent records for a mass properties analysis. (This TMX is a verified expansion of the LaRC working paper "Computer Program for Determining Mass Properties of a Composite Body", by Phillip J. Klich and John L. Gilbert dated Oct. 22, 1968.)

SYMBOLS

I	moments and products of inertia
\vec{r}	displacement vector for differential mass
T	kinetic energy
\vec{V}	linear velocity
x', y', z'	rectangular coordinate component displacement vectors
x, y, z	rectangular coordinate system displacement vectors
$\vec{\omega}$	angular velocity
$l_{x'x}$	cosine of angle between x' and x axes
$l_{y'x}$	cosine of angle between y' and x axes
$l_{z'x}$	cosine of angle between z' and x axes
$l_{x'y}$	cosine of angle between x' and y axes

$l_{y'y}$	cosine of angle between y' and y axes
$l_{z'y}$	cosine of angle between z' and y axes
$l_{x'z}$	cosine of angle between x' and z axes
$l_{y'z}$	cosine of angle between y' and z axes
$l_{z'z}$	cosine of angle between z' and z axes

Subscripts

co system coordinates to component center of mass

$\left. \begin{array}{l} xxco \\ yyco \\ zzco \end{array} \right\}$ refer to component axis to which moments of inertia are calculated

$\left. \begin{array}{l} XXCO \\ YYCO \\ ZZCO \end{array} \right\}$ refer to system axis to which the moments of inertia are rotated parallel to the system coordinates

Superscripts

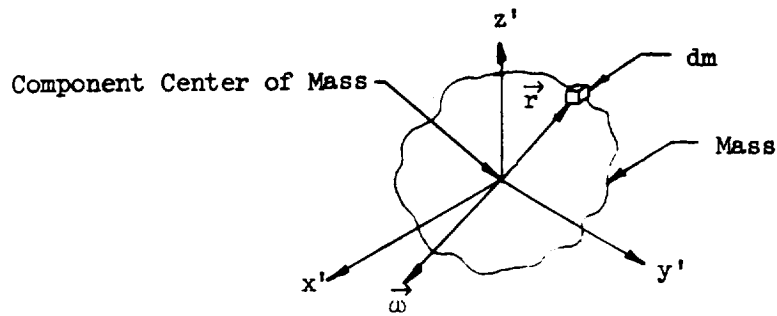
prime denotes component coordinate system

INERTIA EQUATIONS

The mass properties of shapes such as a cylinder, sphere, etc., are easily calculated and therefore were selected as the basic component shapes for handling a system such as a spacecraft structure. Since the component shape mass properties are measured with respect to their respective center of mass, these properties have to be transferred to the system center of mass. The transformation can be made in two steps: first, the component properties are transferred to a system parallel to the system axis, and then transferred by the usual parallel axis theorem. The rotational transformation is derived by using the principle of kinetic energy. An introductory derivation of the moment of inertia, and product of inertia expressions are derived first and then transformation from the component to system coordinates is presented.

Derivation of Inertia Equations

Using the expressions of kinetic energy of a rigid body, the equations of moments of inertia and products of inertia are derived. Consider a component spinning with an angular velocity $\vec{\omega}$ as shown next



The angular kinetic energy can be written as

$$T = \frac{1}{2} \int \vec{V} \cdot \vec{V} \, dm$$

where the velocity is expressed as

$$\vec{V} = \vec{\omega} \times \vec{r}$$

Substituting in the kinetic energy expression gives

$$T = \frac{1}{2} \int (\vec{\omega} \times \vec{r}) \cdot (\vec{\omega} \times \vec{r}) \, dm$$

where

$$\vec{\omega} = i\omega_x' + j\omega_y' + k\omega_z'$$

and

$$\vec{r} = ix' + jy' + kz'$$

Taking the cross product

$$\begin{vmatrix} i & j & k \\ \omega_x' & \omega_y' & \omega_z' \\ x' & y' & z' \end{vmatrix} = i(\omega_y' z' - \omega_z' y') + j(\omega_z' x' - \omega_x' z') + k(\omega_x' y' - \omega_y' x')$$

and performing the dot product results in

$$\begin{aligned}
 & (\omega'_y z')^2 - 2(\omega'_y z')(\omega'_z y') + (\omega'_z y')^2 + (\omega'_z x')^2 - 2(\omega'_z x')(\omega'_x z') \\
 & + (\omega'_x z')^2 + (\omega'_x y')^2 - 2(\omega'_x y')(\omega'_y x') + (\omega'_y x')^2
 \end{aligned}$$

Which upon substituting into kinetic energy equation gives

$$\begin{aligned}
 T = \frac{1}{2} \int & \left[\omega_x'^2 (y'^2 + z'^2) + \omega_y'^2 (x'^2 + z'^2) + \omega_z'^2 (x'^2 + y'^2) \right. \\
 & \left. - 2\omega'_z \omega'_x x' z' - 2\omega'_y \omega'_z y' z' - 2\omega'_x \omega'_y x' y' \right] dm
 \end{aligned}$$

This represents the rotational kinetic energy of one component of a system. Recognizing the definitions of moments and products of inertias and selecting the component coordinate system as the principal axes, we can write

$$T_{\text{comp}} = \frac{1}{2} (I'_x \omega_x'^2 + I'_y \omega_y'^2 + I'_z \omega_z'^2)$$

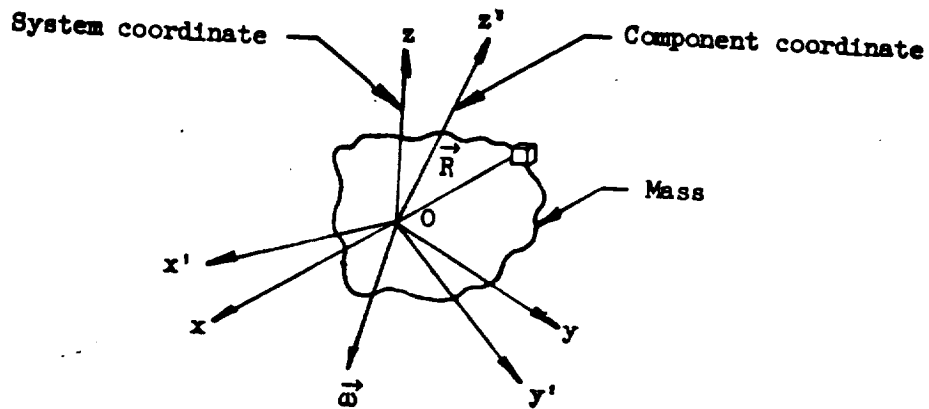
This particular selection of coordinates does not affect the final answers because kinetic energy is constant with regards to the coordinate orientation; however, it does simplify the input data and also reduces computer time.

Rotating From Component Coordinates to System Coordinates

At this point we have the kinetic energy about the component axis system. In the computer program it is at this level that mass properties are computed for the preselected shapes such as the cylinder, sphere, etc.

It is necessary to resolve the component mass properties to an axis system parallel to the system coordinates. Once parallel to this system axis then we can translate to the system center of gravity by the usual parallel axis theorem. The derivation is similar to that presented in reference 5.

In deriving the rotational transformation from the component to the system coordinates the expression for kinetic energy is again used. Given a body rotating with an angular velocity $\vec{\omega}$ we know that its kinetic energy is invariant with regard to the coordinate orientation.



The kinetic energy in the system coordinates is

$$T_{\text{sys}} = \frac{1}{2} \int (\vec{\omega} \times \vec{R}) \cdot (\vec{\omega} \times \vec{R}) \, dm$$

Defining the products of inertia as

$$I_{xz} = \int yz \, dm, \quad I_{yz} = \int yz \, dm, \quad I_{xy} = \int xy \, dm$$

the energy equation becomes

$$T_{\text{sys}} = \frac{1}{2} \left[I_x \omega_x^2 + I_y \omega_y^2 + I_z \omega_z^2 + 2I_{xz} \omega_x \omega_z + 2I_{yz} \omega_y \omega_z + 2I_{xy} \omega_x \omega_y \right]$$

The products of inertia will not necessarily be zero in the system coordinates due to the coordinate rotations and, therefore, the products must be included.

It is necessary to write the angular velocity of the component system coordinates in terms of the system coordinates. For an arbitrary vector it can be written

$$\vec{\omega} = \vec{\omega}'_{\text{comp}} = \vec{\omega}_{\text{system}}$$

$$= i'\omega_{x'} + j'\omega_{y'} + k'\omega_{z'} = i\omega_x + j\omega_y + k\omega_z$$

Performing the dot product gives

$$\omega_{x'} = i' \cdot i\omega_x + i' \cdot j\omega_y + i' \cdot k\omega_z$$

$$\omega_{y'} = j' \cdot i\omega_x + j' \cdot j\omega_y + j' \cdot k\omega_z$$

$$\omega_{z'} = k' \cdot i\omega_x + k' \cdot j\omega_y + k' \cdot k\omega_z$$

Recognizing the direction cosines results in

$$\begin{Bmatrix} \omega_{x'} \\ \omega_{y'} \\ \omega_{z'} \end{Bmatrix} = \begin{bmatrix} l_{x'x} & l_{x'y} & l_{x'z} \\ l_{y'x} & l_{y'y} & l_{y'z} \\ l_{z'x} & l_{z'y} & l_{z'z} \end{bmatrix} \begin{Bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{Bmatrix}$$

Writing the component kinetic energy in matrix algebra

$$T_{\text{comp}} = \frac{1}{2} \{\omega'\}^T [I'] [\omega'] = \frac{1}{2} \left\{ [DC] \{\omega\} \right\}^T [I'] \left\{ [DC] \{\omega\} \right\}$$

where [DC] is the direction cosine matrix shown above. Equating the system and component energy results in

$$T_{\text{system}} = T_{\text{comp}}$$

$$\frac{1}{2} \{\omega\}^T [I] \{\omega\} = \frac{1}{2} \{\omega\}^T [DC]^T [I'] [DC] \{\omega\}$$

and the resulting system inertial matrix is found to be

$$[I] = [DC]^T [I'] [DC]$$

Expanding the matrix equation we get

$$\begin{bmatrix} I_{XXCO} & I_{XYCO} & I_{XZCO} \\ I_{YXCO} & I_{YYCO} & I_{YZCO} \\ I_{ZXCO} & I_{ZYCO} & I_{ZZCO} \end{bmatrix} = \begin{bmatrix} l_{x'x} & l_{y'x} & l_{z'x} \\ l_{x'y} & l_{y'y} & l_{z'y} \\ l_{x'z} & l_{y'z} & l_{z'z} \end{bmatrix} \begin{bmatrix} I'_x & 0 & 0 \\ 0 & I'_y & 0 \\ 0 & 0 & I'_z \end{bmatrix} \begin{bmatrix} l_{x'x} & l_{x'y} & l_{x'z} \\ l_{y'x} & l_{y'y} & l_{y'z} \\ l_{z'x} & l_{z'y} & l_{z'z} \end{bmatrix}$$

The direction cosines are determined by the method presented in appendix A.

Transferring by Parallel Axis Theorem

Now that the moments of inertia are in a system parallel to the system coordinates, we now translate by the parallel axis theorem

$$I_{XX} = I_{XXCO} + m[(Y_{co} - \bar{Y})^2 + (Z_{co} - \bar{Z})^2]$$

$$I_{YY} = I_{YYCO} + m[(X_{co} - \bar{X})^2 + (Z_{co} - \bar{Z})^2]$$

$$I_{ZZ} = I_{ZZCO} + m[(X_{co} - \bar{X})^2 + (Y_{co} - \bar{Y})^2]$$

$$I_{XY} = I_{XYCO} + m[(X_{co} - \bar{X})(Y_{co} - \bar{Y})]$$

$$I_{XZ} = I_{XZCO} + m[(X_{co} - \bar{X})(Z_{co} - \bar{Z})]$$

$$I_{YZ} = I_{YZCO} + m[(Y_{co} - \bar{Y})(Z_{co} - \bar{Z})]$$

Where I_{XXCO} , I_{YYCO} , and I_{ZZCO} are the component moments of inertia rotated parallel to the system coordinates, and \bar{X} , \bar{Y} , \bar{Z} are the system center of mass coordinates and X_{co} , Y_{co} , Z_{co} are the coordinates of the component center of mass.

COMPUTER PROGRAM

This computer program is written in Fortran IV computer language. All names and descriptions are assigned in the first part of the program. Thirteen sections have been written using 13 common shapes usually found in spacecraft. The program is directed by the input data which singles out the section or shape factor desired to be used through the "go to" statement. The operation of the program is illustrated in figure 1 with a computer flow diagram.

The input data for each item is listed on two data cards. The basic input for each item will vary depending on the shape factor used. Each shape factor with the necessary data is discussed in the input data instructions.

After the data cards are supplied to the program the following operations are performed. The component mass properties are first printed with the moments of inertia about the component axis rotated parallel to the system coordinates. These mass properties are transferred to the system center of gravity and the following are computed: System weight, inertias about the system center of gravity, inertias about the origin, center of gravity of the system and products of inertias of the system. Based on this generated information, inertias about the system principal axes and their location is subsequently computed.

A listing of the computer program is found in appendix B.

Selection of Coordinate Points

The selection of points "i" and "j" determines the length of the member as well as the first three direction cosines. Point "k" is required to calculate the other six direction cosines. Shown in figure 2 are the two coordinate systems used in this program. Point "i" locates the system coordinates (X_i , Y_i , Z_i) for the origin of the component axes and point "j" determines the direction of the "x" axis of the component coordinates.

In order to determine the directions of the "y" and "z" axes, point "k" is required. This point can be anywhere in the x-y plane. If it is omitted, then the program automatically positions the "y" axis parallel to the X-Y plane. For a body of revolution point "k" is not required. The following figures (2(a) and 2(b)) describe points i, j, and k.

The main deck is referred to as the computer program without the necessary data. It is always necessary to have a 789 card following the main deck and a 789 card, then a 6789 card following the data. The 6789 card separates one program from another. A description of these cards and their formats follows figure 2.

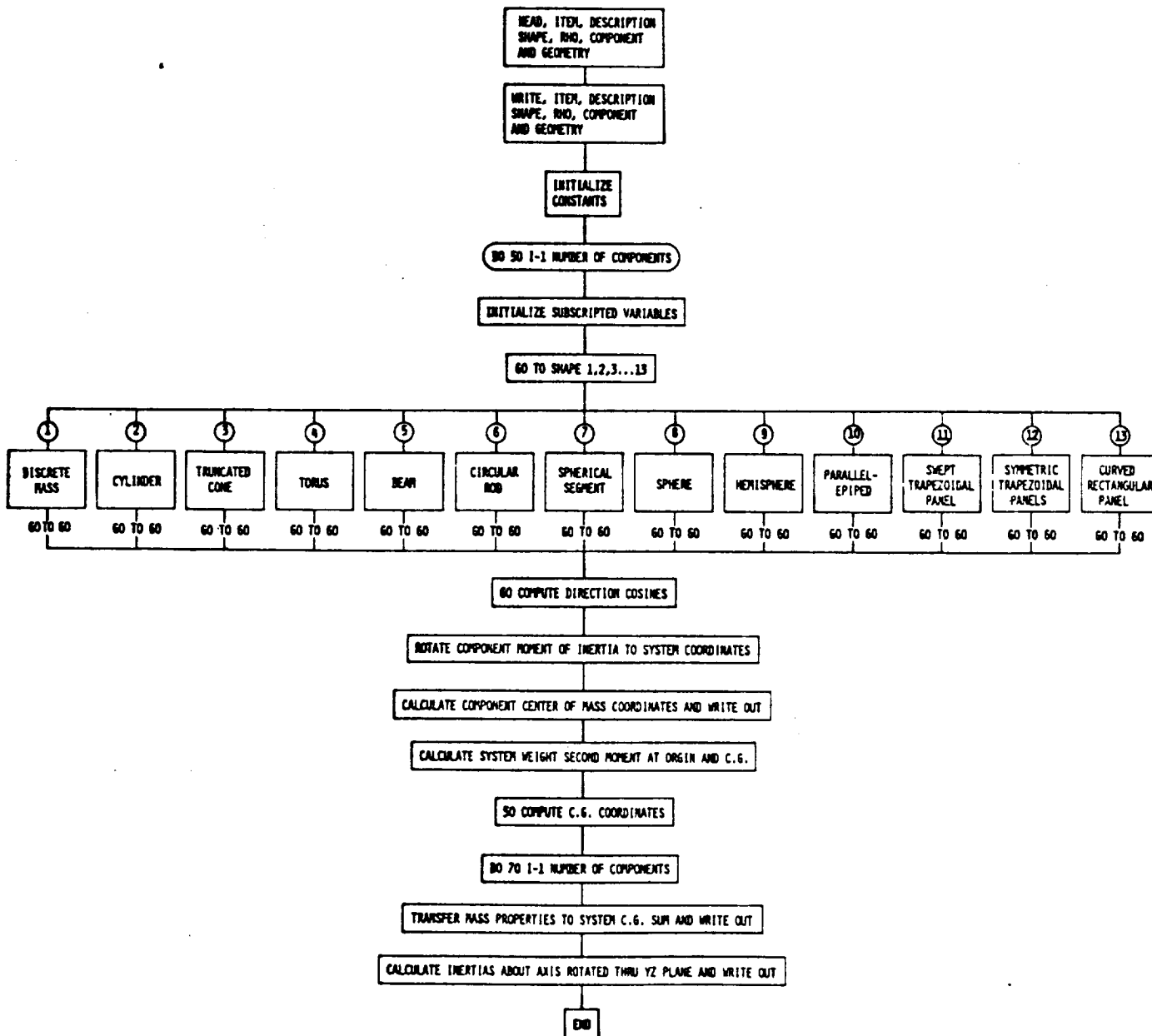
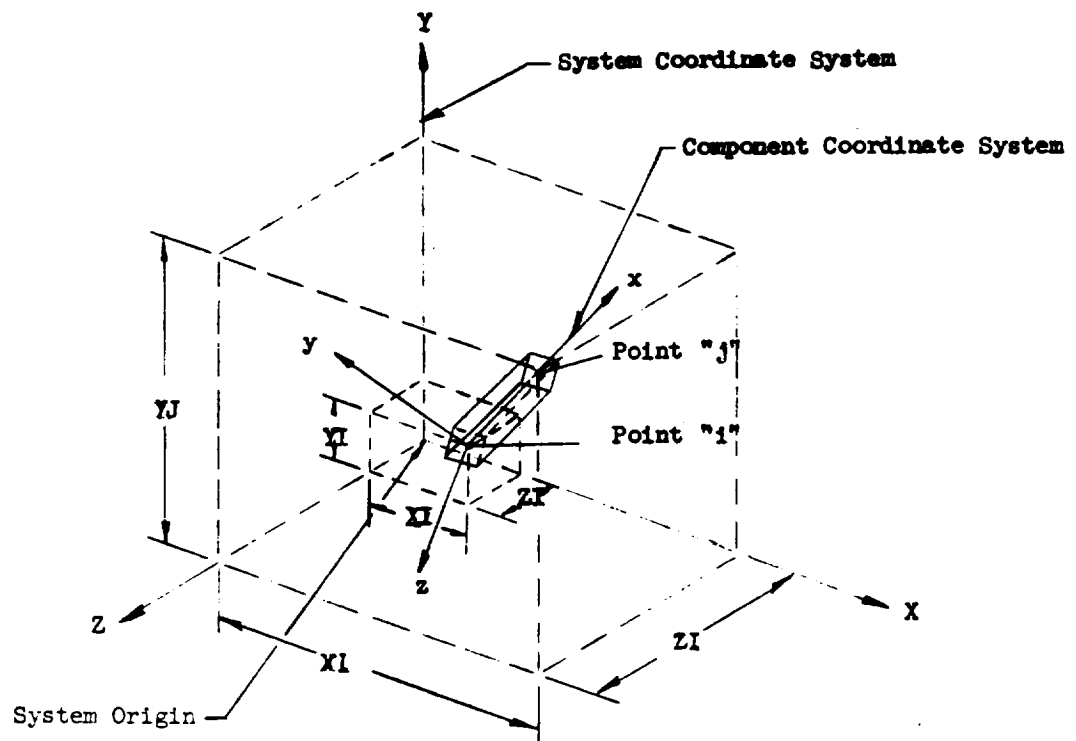
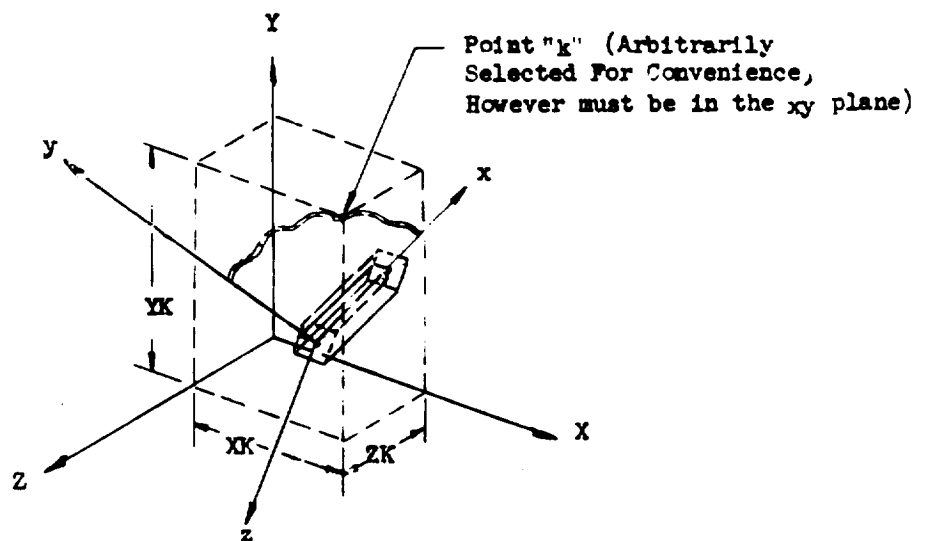


FIGURE 1 - COMPUTER FLOW DIAGRAM



LOCATION OF POINTS "i" AND "j"

Figure 2(a)



LOCATION OF POINTS "k"

Figure 2(b)

1st Data Card

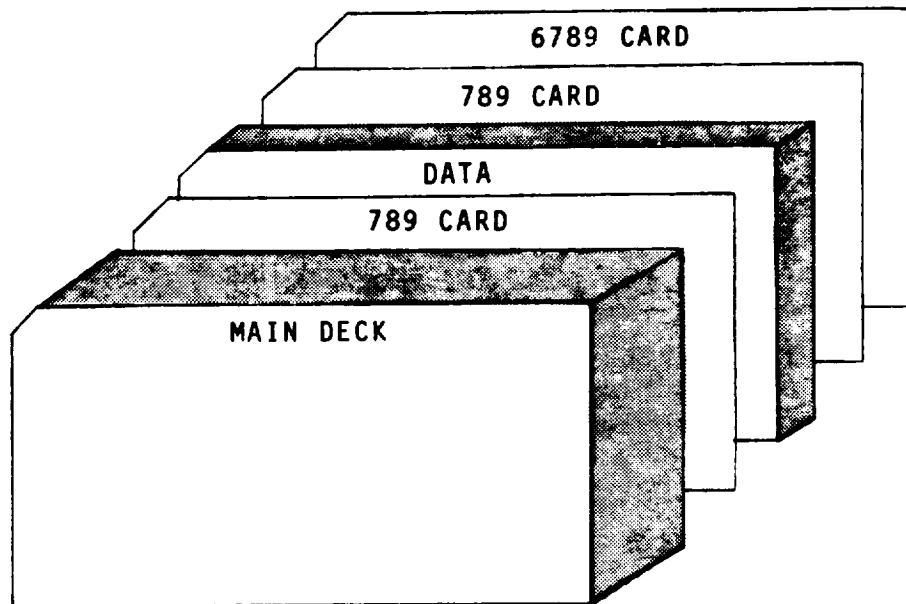
Item No.	I3 format Columns 1 through 3
Description	2A9 format Columns 4 through 21
Shape	I2 format Columns 22 and 23
Weight or density	F9.4 format Columns 24 through 32
A	F8.3 format Columns 33 through 40
B	F8.3 format Columns 41 through 48
C	F8.3 format Columns 49 through 56
D	F8.3 format Columns 57 through 64
F	F8.3 format Columns 65 through 72

It is to be noted that the input data variables A, B, C, D, and F can be geometric dimensions, cross-sectional areas, area moments of inertia, and mass moment of inertia.

2nd Data Card

XI	F8.3 format Columns 1 through 8
YI	F8.3 format Columns 9 through 16
ZI	F8.3 format Columns 17 through 24
XJ	F8.3 format Columns 25 through 32
YJ	F8.3 format Columns 33 through 40
ZJ	F8.3 format Columns 41 through 48
XK	F8.3 format Columns 49 through 56
YK	F8.3 format Columns 57 through 64
ZK	F8.3 format Columns 65 through 72

ORDER OF CARDS FOR COMPUTER PROGRAM



DATA CARDS AND FORMATS (INPUT DATA FOR EACH ITEM)

ITEM NO.		SHAPE					
DESCRIPTION		WT. OR DENSITY	A	B	C	D	F
13	2A9	12	FB.4	FB.3	FB.3	FB.3	FB.3
FORTRAN STATEMENT							
1ST DATA CARD							

XI	VI	ZI	XJ	VJ	ZJ	XK	YK	ZK
FB.3	FB.3	FB.3	FB.3	FB.3	FB.3	FB.3	FB.3	FB.3
FORTRAN STATEMENT								
2ND DATA CARD								

ORIGINAL PAGE IS
OF POOR QUALITY.

Component shapes in the existing programs are:

Shape 1	Discrete Mass
Shape 2	Cylinder
Shape 3	Truncated Cone
Shape 4	Torus
Shape 5	Beam (Arbitrary Cross Section)
Shape 6	Circular Road (Arbitrary Cross Section)
Shape 7	Spherical Segment
Shape 8	Sphere
Shape 9	Hemisphere
Shape 10	Parallelepiped
Shape 11	Swept Trapezoidal Panel
Shape 12	Symmetric Swept Trapezoidal Panels
Shape 13	Curved Rectangular Panel

This program can easily be modified to include additional shapes.

Frequently where precise weights of components are known it is more convenient to input this weight rather than an average density which would have to be calculated. The value of .4 has been chosen as the limiting value for inputting Rho as density (lb/cu in). A value greater than .4 is used as total weight (lbs). In some cases referred to as "thin wall" the total weight of the component must be input. The degree of flexibility for inputting various shapes can be determined from the "shape data input instructions".

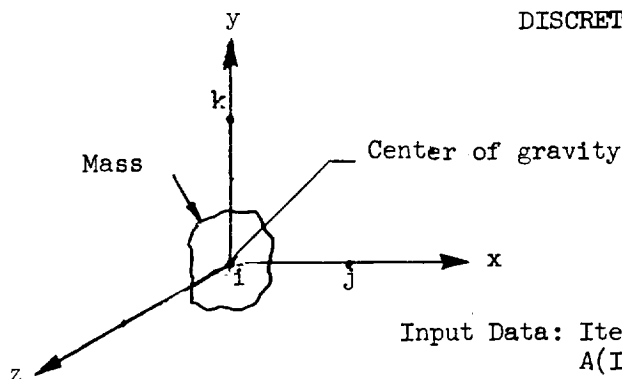
The program can be used to determine the mass properties of a component with hollows or voids. The component is treated as a standard shape with the hollows or voids included as solid material. The hollows or voids are then input as standard shapes having negative values for weight or density. The program will compute the actual weight, center of gravity, and moments and products of inertia of the component.

In the following programs for a variety of shapes the inertias about the x, y and z component axes are represented respectively by computer symbols IXXCG, IYYCG and IZZCG. For some of the shapes expressions for the moments of inertia are given; however, for the more complex shapes they are omitted but can be found by referring to the computer program in Appendix B.

Note that component axes have been located so the x axis is a principal axis and the y and z axes either coincide with, or are parallel to principal axes. Any other than the aforementioned x, y, and z axes locations will incur an error in the main computer program with shape no. 11 being the sole exception. In this instance, the x, y, and z axes locations were chosen for convenience in accordance with the "shape data input instruction" and the x axis is rotated to a principal axis by the program.

SHAPE DATA INPUT INSTRUCTION

SHAPE 1 DISCRETE MASS



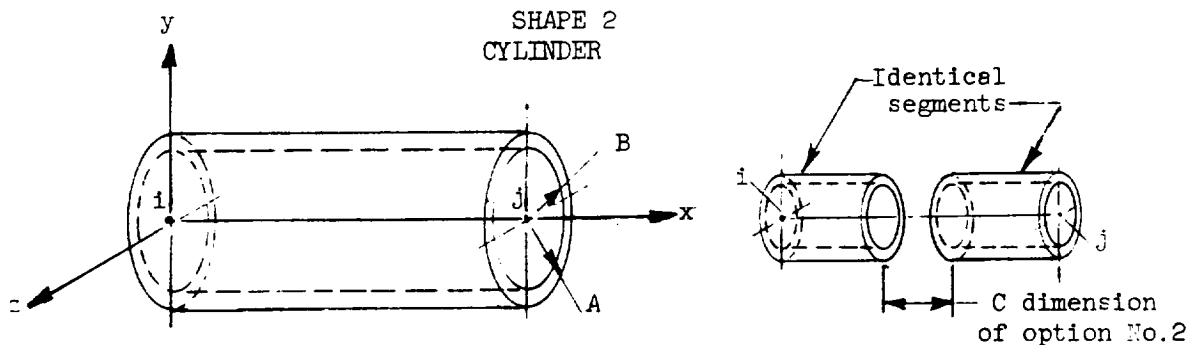
$I_{XXCG} = A$
 $I_{YYCG} = B$
 $I_{ZZCG} = C$

Input Data: Item, Description, Shape, RHO (wt.)
 $A(I_{xx})$, $B(I_{yy})$, $C(I_{zz})$
 XI , YI , ZI , XJ , YJ , ZJ , XK , YK , ZK

OPTIONS AVAILABLE

- No.1 Input data just as indicated above.
- No.2 If negligible, inertias $A(I_{xx})$, $B(I_{yy})$ and $C(I_{zz})$ may be omitted, in which case points j and k are not required.

NOTES: Never locate point j at the system origin and if input, $A(I_{xx})$, $B(I_{yy})$ and $C(I_{zz})$ must be inertias about principal axes in slug-ft. 2



Input Data: Item, Description, Shape, RHO (density)
 A , B , C (option No. 2 only)
 XI , YI , ZI , XJ , YJ , ZJ

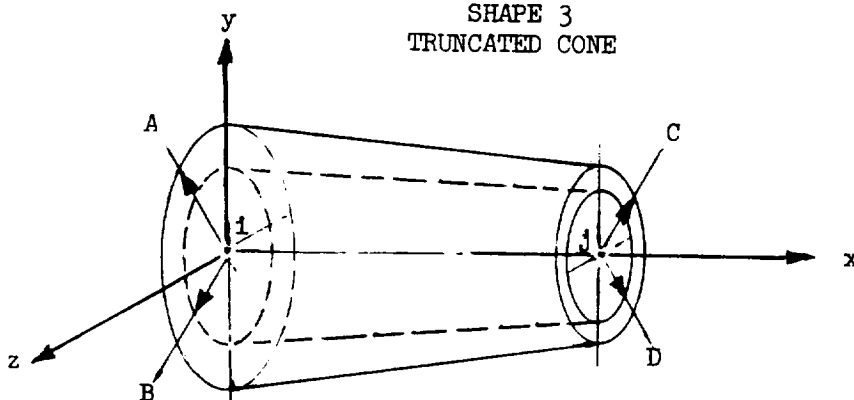
OPTIONS AVAILABLE

- No.1 Input data just as indicated above.
- No.2 Cylinder may be segmented requiring a C value be input.
- No.3 Total weight may be input for RHO in both previous options but only if it is input more than .4 pounds.
- No.4 Input A and B equal and total weight for RHO and program treats shape as a thin-wall cylinder.

NOTES: Point k is not required with any option and density as such must never be input more than .4 pounds per cu. in.

SHAPE DATA INPUT INSTRUCTION

SHAPE 3 TRUNCATED CONE



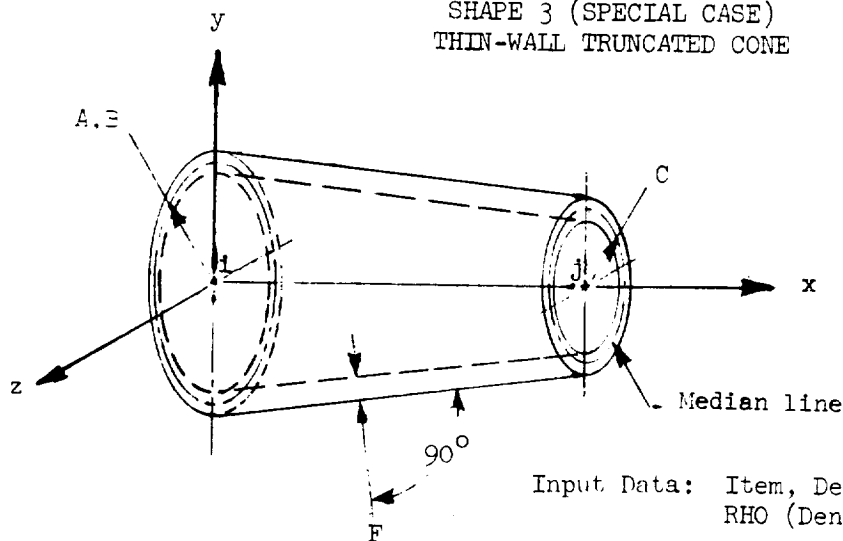
Input Data: Item, Description, Shape, RHO (density), A, B, C, D
 XI, YI, ZI, XJ, YJ, ZJ

OPTIONS AVAILABLE

- No.1 Input data just as indicated above
- No.2 A total weight may be input for RHO but only if it is more than .4 pounds.

NOTES: The values of A minus C and or B minus D must never equal zero.
 Point i is always at the cones Larger end and point k is not required.
 Density as such must never be input greater than .4 lbs./cu.in.

SHAPE 3 (SPECIAL CASE) THIN-WALL TRUNCATED CONE

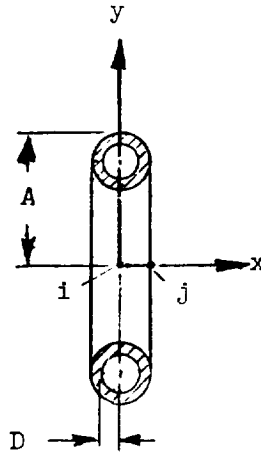
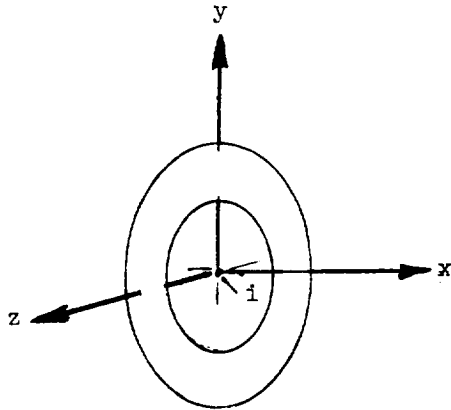


Input Data: Item, Description, Shape
 RHO (Density), A, B, C, F
 XI, YI, ZI, XJ, YJ, ZJ

NOTES: In this case program assumes all mass is concentrated midway between inner and outer surfaces.
 Point i is always at the cone's larger end and point k is not required.

SHAPE DATA INPUT INSTRUCTION

SHAPE 4 TORUS

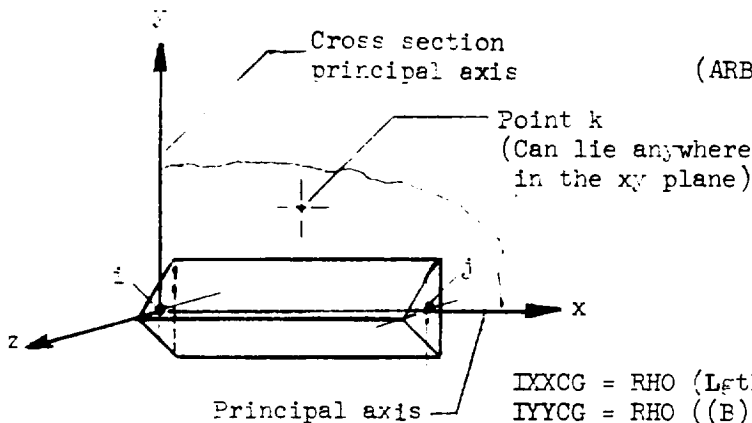


$$\begin{aligned} I_{XXCG} &= Y_{11} - Y_{12} \\ I_{YYCG} &= X_{11} - X_{12} \\ I_{ZZCG} &= I_{YYCG} \end{aligned}$$

Input Data: Item, Description, Shape, RHO (density), A, D
XI, YI, ZI, XJ, YJ, ZJ

NOTE: Point "k" is not required.

SHAPE 5 BEAM (ARBITRARY CROSS SECTION)



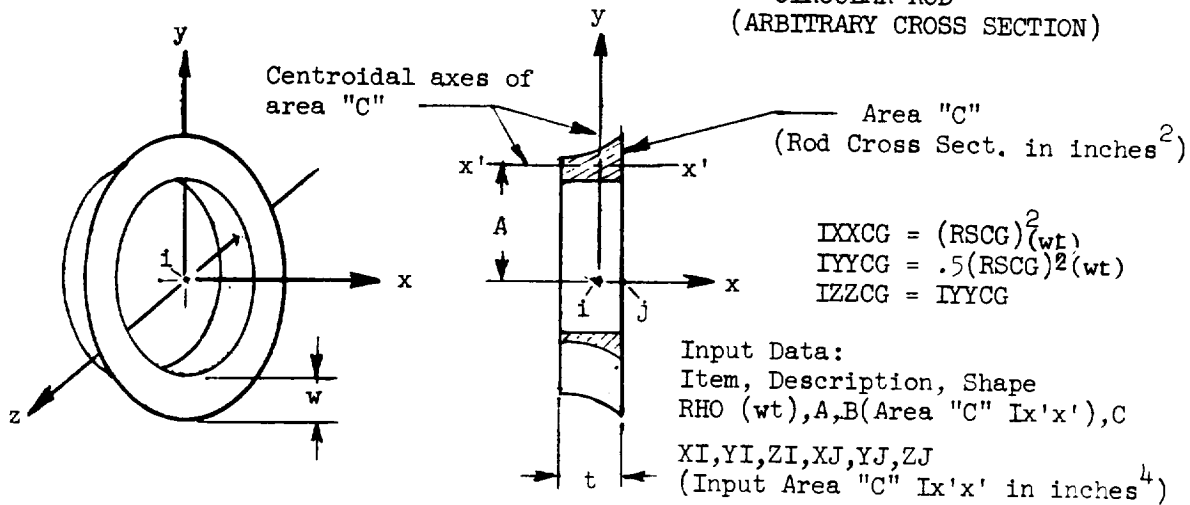
$$\begin{aligned} I_{XXCG} &= RHO (L_{gth})(B + C) \\ I_{YYCG} &= RHO ((B)(L_{gth}) + 0.0833 (A)(L_{gth})^3) \\ I_{ZZCG} &= RHO ((C)(L_{gth}) + 0.0833 (A)(L_{gth})^3) \end{aligned}$$

Input Data: Item, Description, Shape, RHO (density), A (area),
B(I_{yy}), C(I_{zz}) (Area moment of inertia in inches⁴)
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

NOTES: If beam is a body of revolution about the x (centroidal) axis, point k is not required. I_{yy} and I_{zz} are area moments of inertia about principal axes of the beam cross section taken in a plane normal to x axis.

SHAPE DATA INPUT INSTRUCTION

SHAPE 6 CIRCULAR ROD (ARBITRARY CROSS SECTION)

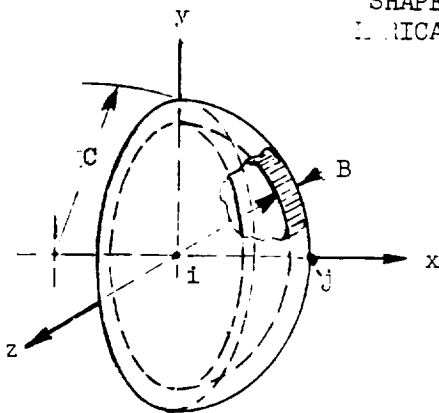


NOTES:

Point "k" is not required and weight instead of density is the required input for RHO.

The solution for this shape is approximate in that it may be as much as one % less than correct if rod dimensions t and w are as much as 25% of dimension A . (Error incurred tends to increase as this % increases)

SHAPE 7 CYLINDRICAL SEGMENT



$$I_{XXCG} = XI1 - XI2$$

$$I_{YYCG} = ACTEMP$$

$$I_{ZZCG} = I_{YYCG}$$

Input Data:
Item, Description, Shape, RHO (density),
B, C (outside radius)

XI, YI, ZI, YJ, YJ, ZJ

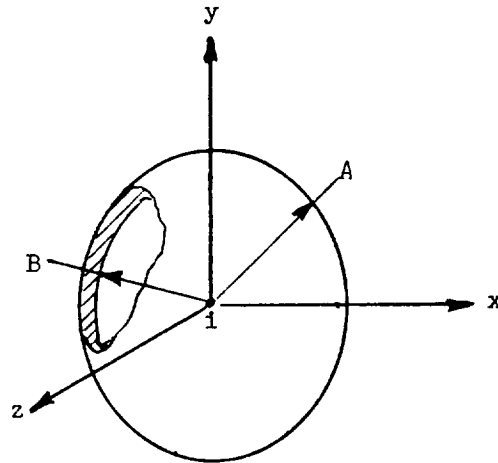
OPTIONS AVAILABLE

- No.1 Input data just as indicated above
- No.2 Total weight may be input for RHO if more than .4 lb.

NOTES:

Point k is not required. When computing as a solid spherical segment, B dim. will become equal the distance between points "i" and "j". Density as such must never be input greater than .4 lbs./cu. in.

SHAPE DATA INPUT INSTRUCTION

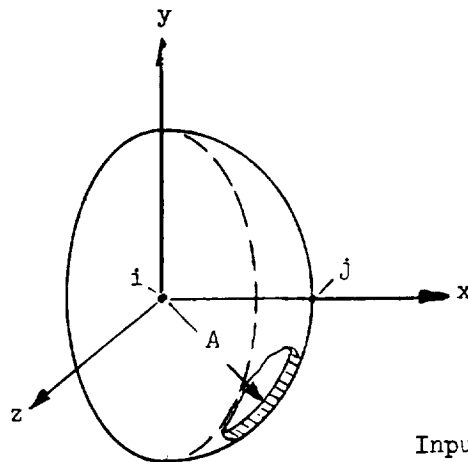


SHAPE 8
SPHERE

IXXCG = XI1-XI2
IYYCG = IXXCG
IZZCG = IXXCG

Input Data: Item, Description, Shape, RHO (density), A,B
XI, YI, ZI

NOTE: Points "j" and "k" are not required. If B = A the program selects thin-wall equations; therefore, weight instead of density should be input for RHO.



SHAPE 9
HEMISPHERE

IXXCG = XI1-XI2
IYYCG = XI3-XI4
IZZCG = IYYCG

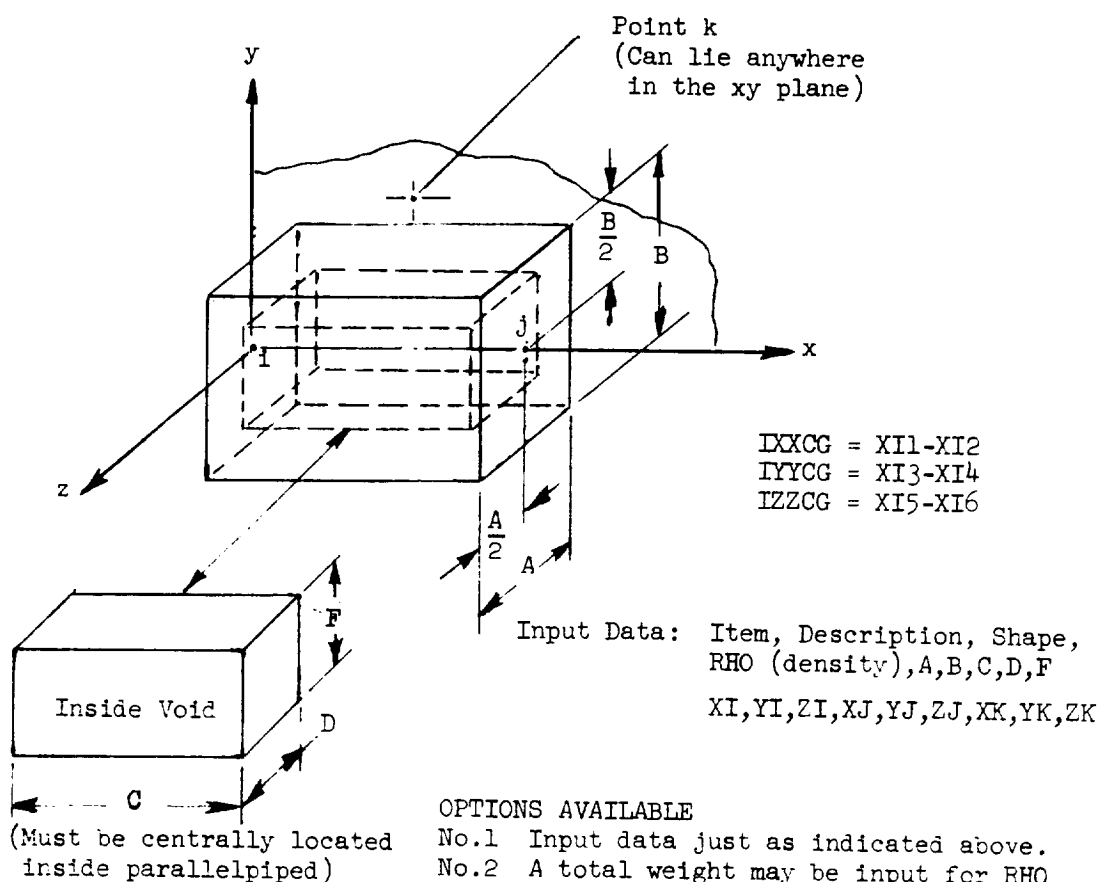
Input Data: Item, Description, Shape,
RHO (density), A

XI, YI, ZI, XJ, YJ, ZJ

NOTES: Point "k" is not required. If A = Lgth (i,j) the program selects thin-wall equations; therefore, weight insted of density should be input for RHO.

SHAPE DATA INPUT INSTRUCTION

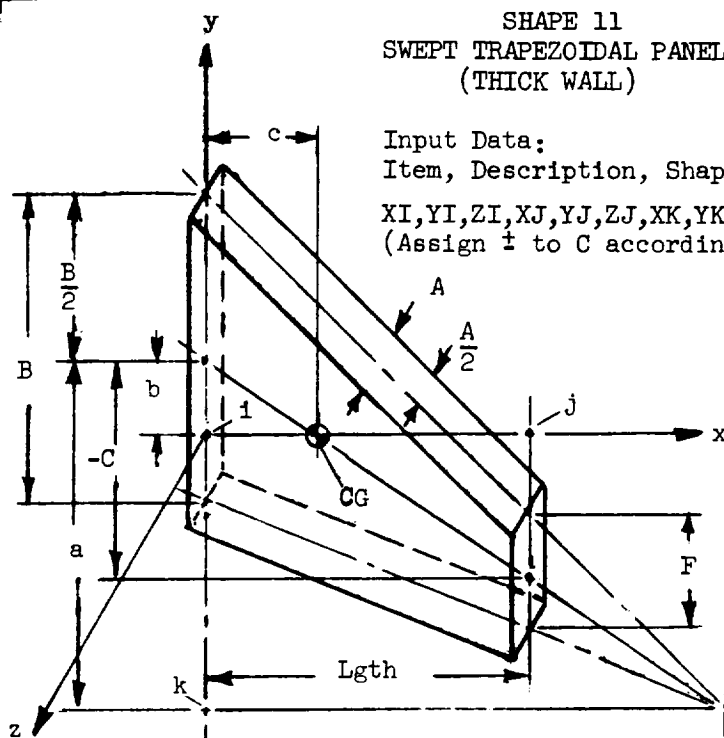
SHAPE 10 PARALLELEPIPED



NOTE: Density as such must never be input greater than .4 lbs./cu.in.

SHAPE DATA INPUT INSTRUCTION

SHAPE 11 SWEEPED TRAPEZOIDAL PANEL (THICK WALL)



Input Data:
Item, Description, Shape, RHO (density), A, B, $\pm C$, F
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK
(Assign \pm to C according to sweep of panel)

Formulas designed to
facilitate input data
determination:

$$a = \frac{C(B)}{B-F}$$

$$b = \frac{C(B+2F)}{3(B+F)}$$

$$c = \frac{Lgth(B+2F)}{3(B+F)}$$

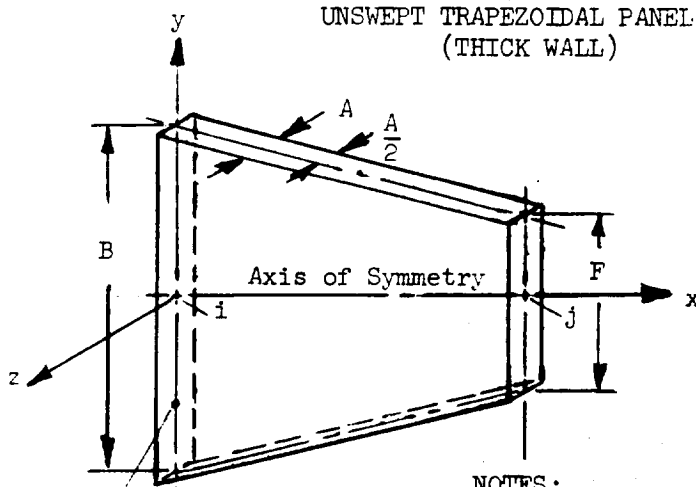
OPTIONS AVAILABLE

- No.1 Input data just as indicated above
- No.2 Total weight may be input for RHO if more than .4 lbs.

NOTES:

Dimension F must never be more than 98% of B. Density as such must never be input greater than .4 lb./cu.in.

SHAPE 11 (SPECIAL CASE) UNSWEPT TRAPEZOIDAL PANEL (THICK WALL)



Input Data:
Item, Description, Shape,
RHO (density), A, B, F
XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

OPTIONS AVAILABLE

- No.1 Input data just as indicated above.
- No.2 Total weight may be input for RHO if more than .4 lbs.

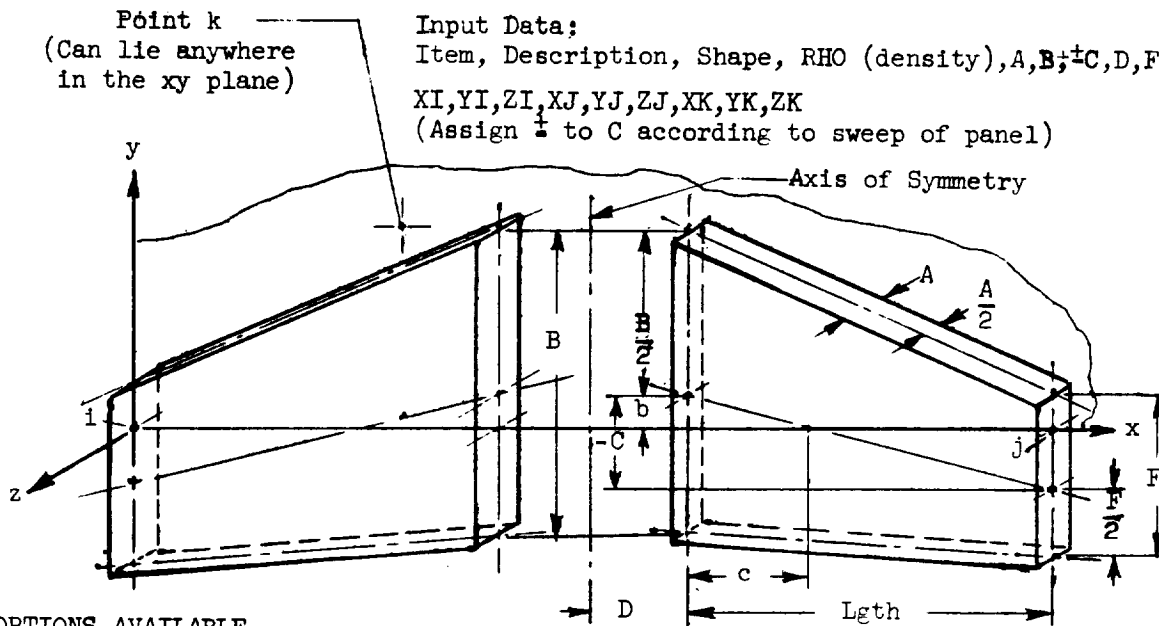
NOTES:

Dimension F must never be more than 98% of B. Density as such must never be input greater than .4 lbs./cu.in.

Point k (Can lie anywhere on the y axis except at point i)

SHAPE DATA INPUT INSTRUCTION

SHAPE 12 SYMMETRIC SWEEPED TRAPEZOIDAL PANELS (THICK WALL)



OPTIONS AVAILABLE

- No.1 Input data just as indicated above.
No.2 Total weight may be input for RHO
if it is more than .4 lb.

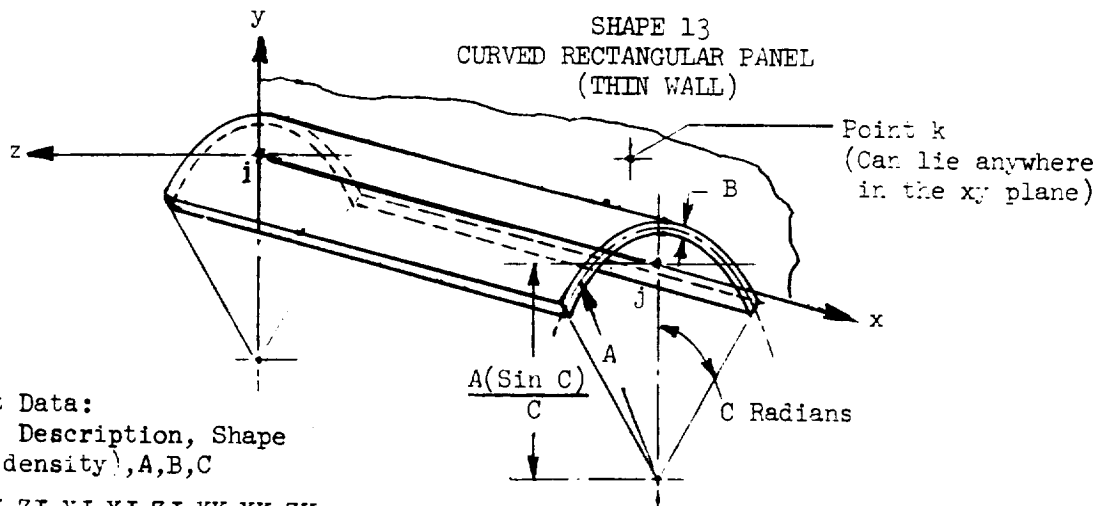
NOTES:

Dimension F must be no more than 98% of B.
Density as such must never be input greater
than .4 lb./cu.in.

Formulas designed to facilitate
input data determination:

$$b = \frac{C(B+2F)}{3(B+F)} \quad c = \frac{Lgth(B+2F)}{3(B+F)}$$

SHAPE 13 CURVED RECTANGULAR PANEL (THIN WALL)



Input Data:

Item, Description, Shape
RHO (density), A, B, C

XI, YI, ZI, XJ, YJ, ZJ, XK, YK, ZK

NOTE: Plane xy is one about which symmetry exists.

EXAMPLE PROBLEMS

Two example problems are presented in order to show required input data. These problems were selected due to the simple calculations involved and thus could be checked by hand calculations.

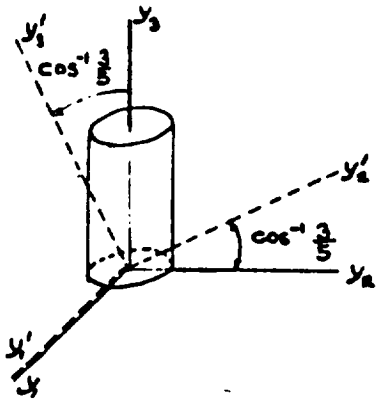
Example Problem 1

Problem 1 was taken from reference 3. It is a cylinder skewed in the y-z plane. The moments of inertia about the system origins are given in this reference and are used for comparison in this paper. It should be noted that "g" (acceleration of gravity) was taken to be 32.0 ft/sec^2 in this reference rather than 32.2 ft/sec^2 (386 in./sec^2). Depending on the value of "g" selected by the program user, the term "cons" has to be changed accordingly.

The axis through the center of the component must always be the x-axis for the computer program. Therefore, the y_3 axis of this problem corresponds with the x-axis of the computer program; the y_1 axis corresponds with the y-axis and the y_2 axis corresponds with the z-axis.

Find moments of inertia I'_{11} , I'_{22} , I'_{33} which corresponds to I_{xxo} , I_{yyo} , I_{zzo} in our coordinate system.

Given: $W = 1 \text{ slug} = 32 \text{ lb}$
 $R = 24 \text{ inches}$
 $L = 36 \text{ inches}$
 $\theta = \cos^{-1} 3/5 = 53^\circ 8'$



Solution: (a) Direction cosines for the transformation are

$$a_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{3}{5} & \frac{4}{5} \\ 0 & -\frac{4}{5} & \frac{3}{5} \end{bmatrix}$$

And numerical values of the I_{ij} are

$$I_{ij} = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad \begin{array}{l} \text{Component} \\ \text{Moment of} \\ \text{Inertia} \end{array}$$

ORIGINAL PAGE IS
OF POOR QUALITY

The results of problem 1 as computed in reference 3 are now given. Hence, from the transformation equations

System Moments of Inertia

$$I'_{11} = a_{11}a_{11}I_{11} + a_{12}a_{12}I_{22} + a_{13}a_{13}I_{33} + 2a_{12}a_{13}I_{23} + 2a_{11}a_{13}I_{13} + 2a_{11}a_{12}I_{12}$$

$$I_{xx} = (1)^2 4 + 0 + 0 + 0 + 0 + 0 = 4 = \text{slug ft}^2 \quad (I_{xx} = I'_{11})$$

$$I'_{22} = a_{21}a_{21}I_{11} + a_{22}a_{22}I_{22} + a_{23}a_{23}I_{33} + 2a_{22}a_{23}I_{23} + 2a_{21}a_{23}I_{13} + 2a_{21}a_{22}I_{12}$$

$$I_{yy} = 0 + \left(\frac{3}{5}\right)^2 4 + \left(\frac{4}{5}\right)^2 2 + 0 + 0 + 0 = \frac{68}{25} = 2.72 \text{ slug ft}^2 \quad (I_{yy} = I'_{22})$$

$$I'_{33} = a_{31}a_{31}I_{11} + a_{32}a_{32}I_{22} + a_{33}a_{33}I_{33} + 2a_{32}a_{33}I_{23} + 2a_{31}a_{33}I_{13} + 2a_{31}a_{32}I_{12}$$

$$I_{zz} = 0 + -\frac{4}{5}^2 4 + \frac{3}{5}^2 2 + 0 + 0 + 0 = \frac{32}{25} = 3.28 \text{ slug ft}^2$$

This problem is now computed using Computer Program NDC702. Shown below are the coordinates of points i, j, and k for problem 1.

$$\begin{array}{l} \text{Point "i"} \end{array} \quad \begin{cases} X_i = 0 \\ Y_i = 0 \\ Z_i = 0 \end{cases}$$

$$\begin{array}{l} \text{Point "j"} \end{array} \quad \begin{cases} X_j = 0 \\ Y_j = 30 \sin 53^\circ \theta' = 28.8 \\ Z_j = 30 \cos 53^\circ \theta' = 21.6 \end{cases}$$

Point "k" Omit since not required for cylinder

Reference is now made to the necessary data cards to compute this problem.

We now have the results of problem 1 using Computer Program

(Note that the gravitational constant used here was 32.166 rather than 32.0 used in ref.)

INPUT DATA LISTED BELOW

ITEM	DESCRIPTION	SHAPE	P40	A	B	C	D	F
1	CYLINDER, TEST	Z	.3200000E+02	20.000	0.000	0.000	0.000	0.000
	HT	VI	ZI	VJ	ZJ	XK	YK	ZK
	0.000	0.000	0.000	0.000	20.000	21.000	0.000	0.000

COMPONENT DATA LISTED BELOW

ITEM	DESCRIPTION	WT	IXCO	IYCO	IZZCO	XC6CO	YC6CO	ZC6CO	IXYCO	IYZCO	IXZCO
1	CYLINDER, TEST	32.00000	1.70000	1.00010	1.03047	0.00000	10.00000	10.00000	0.00000	.11930	0.00000

SYSTEM DATA LISTED BELOW (INCLHS, INERTIALS IN SLUGS FT SQUARED, C.G. IN INCHES, SECOND MOMENTS IN SLUGS FT SQUARED)

SVS HT	IXCO	IYCO	IZZCO	XC6CO	YC6CO	ZC6CO
32.000	1.700	1.000	1.030	0.000	10.000	10.000
IXX	IYY	IZZ	IXY	IYZ	IXZ	
1.70000	1.00010	1.03047	0.00000	0.00000	.11930	

INERTIALS (EIGENVALUES) ABOUT SYSTEM PRINCIPAL AXES WITH AXIS DIRECTION COSINES (EIGENVECTORS) RELATING THE PRINCIPLE AXES TO THE X, Y, AND Z SYSTEM AXES IN THAT SEQUENCE

EIGENVALUE(1) = .17000E+01

EIGENVECTOR(1)

.100000E+01 0. 0.

EIGENVALUE(2) = .17000E+01

EIGENVECTOR(2)

0. -.000000E+00 .000000E+00

EIGENVALUE(3) = .19000E+01

EIGENVECTOR(3)

0. .000000E+00 .000000E+00

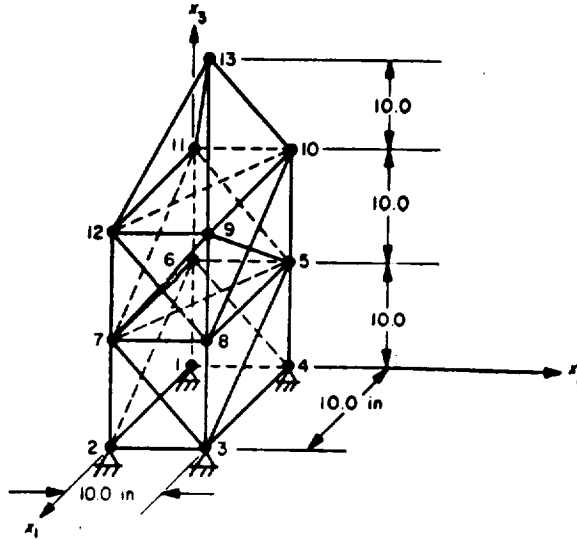
Example Problem 2

Problem 2 was taken from reference 2. In this space structure, the weight has been 'lumped' or concentrated at the joints. In the program being presented, this is not required but is used here only for illustration. To compare moment of inertia, the numbers given in reference 2 should be converted to slug-ft².

$$I_{XX} = I_{YY} - \frac{22038.464}{4.608E3} = 4.783 \text{ slug-ft}^2$$

$$I_{ZZ} = \frac{9000.00}{4.0608F3} = 1.953 \text{ slug-ft}^2$$

Example Problem 2



INPUT:
MEMBER AREAS - 0.01 in^2
WEIGHT AT EACH JOINT - 15 Q lb
MODULUS OF ELASTICITY - 10^{10} psi
SEE THE FIRST TWO PAGES OF THE SAMPLE
PROBLEM FOR DETAILS OF THE INPUT

JOINT COORDINATES				
JOINT	X1	X2	Y1	Y2
1	0.	0.	0.	0.
2	10.00000	0.	0.	0.
3	10.00000	10.00000	0.	0.
4	0.	10.00000	0.	0.
5	0.	10.00000	10.00000	0.
6	0.	0.	10.00000	10.00000
7	10.00000	0.	10.00000	10.00000
8	10.00000	10.00000	10.00000	10.00000
9	10.00000	20.00000	20.00000	20.00000
10	0.	10.00000	20.00000	20.00000
11	0.	0.	20.00000	20.00000
12	10.00000	0.	20.00000	20.00000
13	10.00000	10.00000	30.00000	30.00000

The results of problem 2 as computed in reference 6 are now given.

TOTAL WEIGHTS	195.000	195.000	195.000	0.	0.	0.
CENTER OF WEIGHT	X = 5.000,	Y = 5.000,	Z = 11.538			
WEIGHT MOMENTS OF INERTIA ABOUT CENTER OF WEIGHT						
IXX = 22036.462,	IYY = 22036.462,	IZZ = 9000.000,	IXY = -0.	IYZ = 0.000,	IYZ = 0.000	

NOTE: Inertias are given in pound inches squared and must be converted to compare with results program yields.

This problem is now computed using Computer Program ND0702

ORIGINAL PAGE IS
OF POOR QUALITY

INPUT DATA LISTED BELOW

ITEM	DESCRIPTION	SHAPE	RHO	A	B	C	D	E
1	CODE 1	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
2	CODE 2	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
3	CODE 3	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
4	CODE 4	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
5	CODE 5	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
6	CODE 6	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
7	CODE 7	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
8	CODE 8	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
9	CODE 9	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
10	CODE 10	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
11	CODE 11	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
12	CODE 12	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000
13	CODE 13	1	.15000000E+02	0.000	0.000	0.000	0.000	0.000

COMPONENT DATA LISTED BELOW

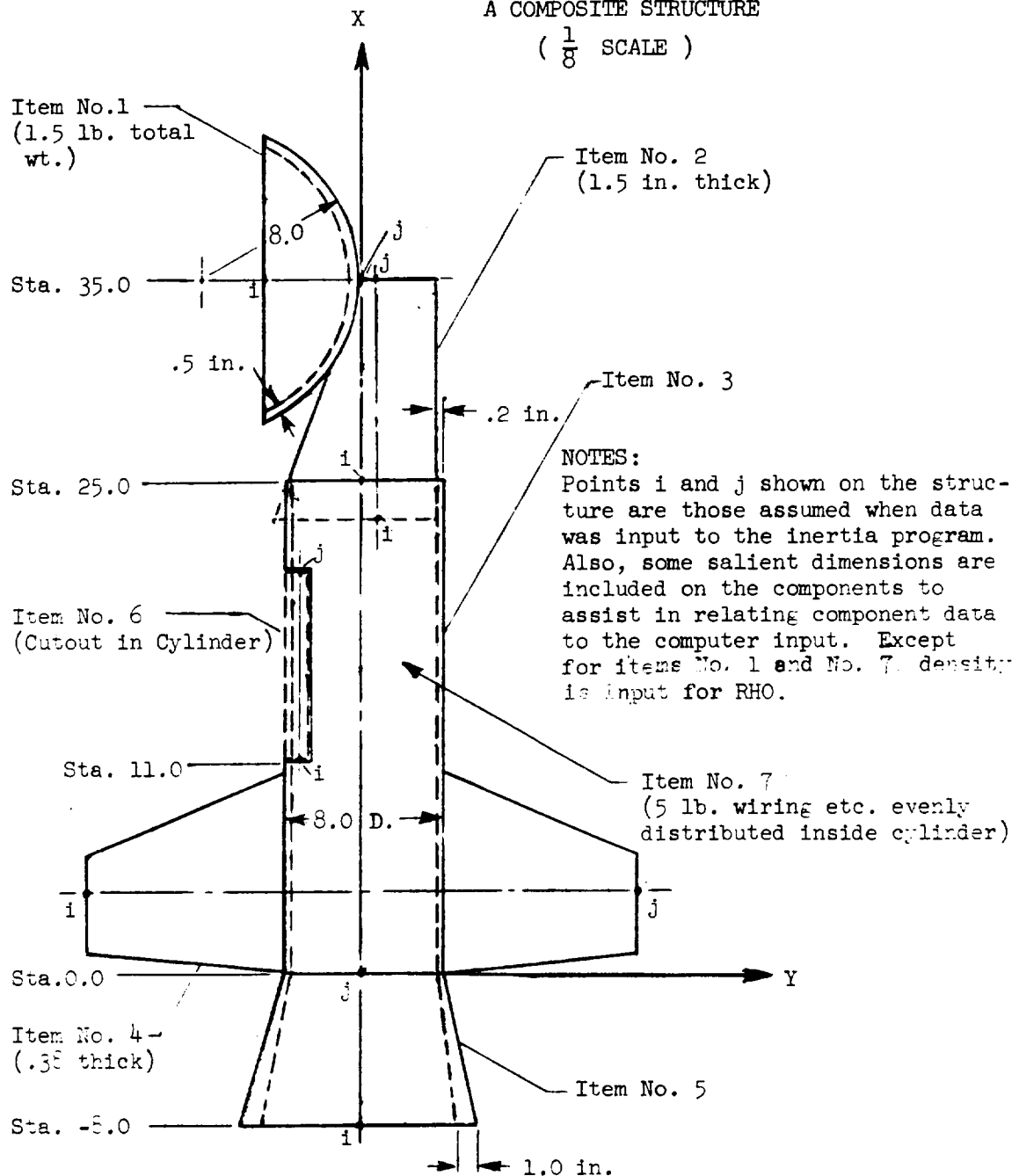
ITEM	DESCRIPTION	PI	IXCO	IYCO	IZCO	XCOC	YCOC	ZCOC	IXCO	IYCO	IZCO
1	CODE 1	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2	CODE 2	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	0.00000	0.00000	0.00000	0.00000
3	CODE 3	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	0.00000	0.00000	0.00000	0.00000
4	CODE 4	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	0.00000	0.00000	0.00000	0.00000
5	CODE 5	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	10.00000	0.00000	0.00000	0.00000
6	CODE 6	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	10.00000	0.00000	0.00000	0.00000
7	CODE 7	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	10.00000	0.00000	0.00000	0.00000
8	CODE 8	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	10.00000	0.00000	0.00000	0.00000
9	CODE 9	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	20.00000	0.00000	0.00000	0.00000
10	CODE 10	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	20.00000	0.00000	0.00000	0.00000
11	CODE 11	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	20.00000	0.00000	0.00000	0.00000
12	CODE 12	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	20.00000	0.00000	0.00000	0.00000
13	CODE 13	15.00000	0.00000	0.00000	0.00000	5.00000	5.00000	50.00000	0.00000	0.00000	0.00000

SYSTEM DATA LISTED BELOW (X-COORD, INERTIAL; Y-COORD, X-COORD, SECOND COMPONENT; Z-COORD, Y-COORD)

ITEM	DESCRIPTION	PI	IXCO	IYCO	IZCO	XCOC	YCOC	ZCOC
1	CODE 1	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
2	CODE 2	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	0.00000
3	CODE 3	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	0.00000
4	CODE 4	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	0.00000
5	CODE 5	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	10.00000
6	CODE 6	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	10.00000
7	CODE 7	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	10.00000
8	CODE 8	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	10.00000
9	CODE 9	15.00000	0.00000	0.00000	0.00000	10.00000	10.00000	20.00000
10	CODE 10	15.00000	0.00000	0.00000	0.00000	0.00000	10.00000	20.00000
11	CODE 11	15.00000	0.00000	0.00000	0.00000	0.00000	0.00000	20.00000
12	CODE 12	15.00000	0.00000	0.00000	0.00000	10.00000	0.00000	20.00000
13	CODE 13	15.00000	0.00000	0.00000	0.00000	5.00000	5.00000	50.00000

Note: No principal axis definition is included since they will lie on axis of symmetry.

EXAMPLE PROBLEM 3
A COMPOSITE STRUCTURE
($\frac{1}{8}$ SCALE)



The computer output relative to example problem 3 follows and includes, the data input, the computer calculated component data, the summed data and lastly, the inertias about the system principal axes and the individual axis locations.

INPUT DATA LISTED BELOW

ITEM	DESCRIPTION	SHAPE	RHO	A	B	C	D	F
1	ANTENNA	7	.19000000E+01	0.000	0.000	0.000	0.000	0.000
2	DIELECTRIC BRKT	11	.15000000E+00	1.500	0.000	-2.000	0.000	3.000
3	BODY CYL	2	.10000000E+00	4.000	1.000	0.000	0.000	0.000
4	PINS	12	.66000000E-01	.300	10.000	-1.300	4.000	5.000
5	FLARED ADAPTER	3	.10000000E+00	0.000	5.000	4.030	3.000	0.000
6	CYL SEC VOID	13	-.10000000E+00	3.000	.200	1.000	0.000	0.000
7	WIRING ETC VOL	2	.30000000E+01	3.000	0.000	0.000	0.000	0.000

COMPONENT DATA LISTED BELOW

ITEM	DESCRIPTION	WT	IXXC0	IYYC0	IZZC0	XC6C0	YC6C0	ZC6C0	IXYC0	IYZC0	IIZC0
1	ANTENNA	1.30000	.00937	.00951	.02537	35.00000	-2.62146	0.00000	0.00000	0.00000	3.00000
2	DIELECTRIC BRKT	16.20000	.01442	.04039	.05349	28.20000	.64000	0.00000	-.00793	0.00000	3.00000
3	BODY CYL	12.25221	.04026	.15790	.15790	12.50000	0.00000	0.00000	0.00000	0.00000	3.00000
4	PINS	3.76200	.06444	.00439	.04881	4.33000	0.00000	0.00000	0.00000	0.00000	3.00000
5	FLARED ADAPTER	14.71107	.07771	.05261	.05261	-5.07517	0.00000	0.00000	0.00000	0.00000	3.00000
6	CYL SEC VOID	-1.56000	-.00150	-.00420	-.00291	16.00000	-3.29000	0.00000	0.00000	0.00000	0.00000
7	WIRING ETC VOL	3.00000	.02779	.06012	.06012	12.50000	0.00000	0.00000	0.00000	0.00000	0.00000

SYSTEM DATA LISTED BELOW (WT=LBS, INERTIAS=SLUGS FT SQUARED, C.G.=INS, SECOND MOMENT=SLUG FT SQUARED)

SYS WT	IXX0	IYY0	IZZ0	IXX4	YBAR	ZBAR
91.845	.209	4.091	4.166	12.372	.223	0.000
128	IYY	IZZ	IXY	IZY	IZZ	
.20797	2.37764	2.45180	-.02017	0.00000	0.00000	

INERTIAS (EIGENVALUES) ABOUT SYSTEM PRINCIPAL AXES WITH AXIS DIRECTION COSINES (EIGENVECTORS) RELATING THE PRINCIPAL AXES TO THE X, Y, AND Z SYSTEM AXES IN THAT SEQUENCE

EIGENVALUE(1) = .20760E+00

EIGENVECTOR(1)

.999916E+00 .129820E-01 0.

EIGENVALUE(2) = .23760E+01

EIGENVECTOR(2)

-.129820E-01 .999916E+00 0.

EIGENVALUE(3) = .24518E+01

EIGENVECTOR(3)

0. 0. .100000E+01

CONVERSION TO INTERNATIONAL SYSTEM OF UNITS

<u>CONVERT FROM</u>	<u>TO</u>	<u>MULTIPLY BY</u>
INCHES	METERS	.025 400
INCHES ²	METERS ²	.000 645 160
FEET	METERS	.304 800
FEET ²	METERS ²	.092 903 040
POUNDS	KILOGRAMS	.453 592 370
POUNDS/INCHES ³	KILOGRAM/METER ³	27 679.905
POUNDS/FEET ³	KILOGRAM/METER ³	16.018 463
SLUG	KILOGRAMS	14.593 9029
SLUG FEET ²	KILOGRAM METER ²	1.355 8179

CONCLUSIONS

A computer program for determining mass properties of a rigid structure is presented. The structure is broken down into preselected shapes with known properties, and input data are supplied to completely describe each shape. For complicated structures skewed in space, this program offers a practical solution to a tedious and time-consuming task. It is also practical to use this program for problems that involve repetitious or lengthy calculations.

REFERENCES

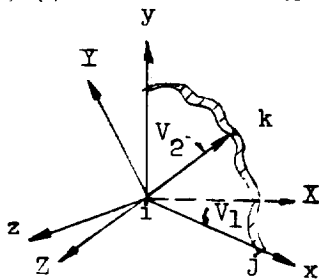
1. Thomson, W. T.: Introduction to Space Dynamics, 1961.
2. Meyers, F. T.: Weight Handbook, Volume 1, 1944.
3. Pletta, D. H., and Frederick, D.: Engineering Mechanics, 1963.
4. Wada, B.: Stiffness Matrix Structural Analysis, TN-32-774 (JPL).
5. Greenwood, D. T.: Principles of Dynamics, 1965.

ORIGINAL PAGE IS
OF POOR QUALITY

APPENDIX A

DIRECTION COSINE DERIVATION

The direction cosines are used to transform the properties from the component coordinate system x, y, z to system coordinates X, Y, Z . Three points (i, j, k) shown in the figure below, define the vectors \vec{V}_1 and \vec{V}_2



Vector \vec{V}_1 is arbitrarily selected to be coincident with the "x" axis. A unit vector on this axis can be written

$$\vec{1}_x = \frac{\vec{V}_1}{|\vec{V}_1|}$$

Taking the vector cross product of \vec{V}_1 with \vec{V}_2 and dividing by the resulting magnitude gives a unit vector on the z axis

$$\vec{1}_z = \frac{\vec{V}_1 \times \vec{V}_2}{|\vec{V}_z|}$$

Similarly, a unit vector on the "y" axis is found from

$$\vec{1}_y = \vec{1}_z \times \vec{1}_x$$

The direction cosines are the X, Y, Z , components of the unit vectors on the x, y, z axes.

The direction cosine for the x axis is written as

$$\begin{aligned} LX &= (XJ - XI) / LGTH \\ MX &= (YJ - YI) / LGTH \\ NX &= (ZJ - ZI) / LGTH \end{aligned}$$

where the length is

$$LGTH = ((XJ - XI)^2 + (YJ - YI)^2 + (ZJ - ZI)^2)^{1/2}$$

The vector \vec{V}_2 can be written

$$\begin{aligned} T1 &= XK - XI \\ T2 &= YK - YI \\ T3 &= ZK - ZI \end{aligned}$$

A vector on the z axis is found by taking the vector cross product of $\vec{1}_x$ and \vec{V}_2

$$\begin{aligned} LZ &= MX \cdot T3 - T2 \cdot NX \\ MZ &= NX \cdot T1 - T3 \cdot LX \\ NZ &= T2 \cdot LX - T1 \cdot MX \end{aligned}$$

The length is

$$T4 = (LZ^2 + MZ^2 + NZ^2)^{1/2}$$

Normalizing to get a unit vector

$$LZ \rightarrow LZ/T4; MZ \rightarrow MZ/T4; NZ \rightarrow NZ/T4$$

The unit vector on the y axis has a magnitude of one and is determined by the vector cross product $\vec{1}_z$ and $\vec{1}_x$

$$\begin{aligned} LY &= MZ \cdot NX - MX \cdot NZ \\ MY &= NZ \cdot LX - NX \cdot LZ \\ NY &= MX \cdot LZ - LX \cdot MZ \end{aligned}$$

Writing the nine terms in matrix form, we get

$$[DC] = \begin{bmatrix} LX & MX & NX \\ LY & MY & NY \\ LZ & MZ & NZ \end{bmatrix}$$

APPENDIX 8

```

PROGRAM NAME      (INPUT,OUTPUT,TAPES=INPUT,TAPF6=OUTPUT)

C   LATEST DIRCOS IN MAIN DECK AND 13 SHAPES 12/20/72

    DIMENSION ITEM(200),SHAPE(200),RHO(200),A(200),R(200),C(200),
1   D(200),F(200),      XI(200),YI(200),ZI(200),XJ(200),YJ(200),ZJ(2
200),XK(200),YK(200),ZK(200),IXXCG(200),IYYCG(200),IZZCG(200),IXYCG
3(200),IYZCG(200),IXZCG(200),XCG(200),YCG(200),ZCG(200),XL(200),YL(
4200),ZL(200),IXXCO(200),IYYCO(200),IZZCO(200),IXYCO(200),IXZCO(200
5),IYZCO(200),DES(3,200),W(200),D(200),P(200),ARR(3,3),F(3),CRR(3)
    REAL IXX,IYY,IZZ,IXXCG,IYYCG,IZZCG,IXYCG,IXZCG,
1IYZCG,LX,MX,NX,LY,MY,NY,LZ,MZ,NZ,IXY,IXZ,IYZ,LGTH,IXXCO,IYYCO,
2IZZCO,IXYCO,IXZCO,IYZCO,IXXO,IYYO,IZZO,KI,KII,KIII,IXYK,IXYCGO
    INTEGER SHAPE
    I=0
1010 I=I+1
    105 FORMAT(I5)
        READ(5,101) ITEM(I),DES(1,I),DES(2,I),SHAPE(I),RHO(I),A(I),R(I),
        IC(I),D(I),F(I),      XI(I),YI(I),ZI(I),XJ(I),YJ(I),ZJ(I),XK(I),
        YK(I),ZK(I)
101 FORMAT (I3,2A9,I2,F9.4,5F8.2/9F8.2)
    IF(FOF(5))1009,1010
1009 IMAX=I-1
    WRITE(6,271)
271 FORMAT(1H0* INPUT DATA LISTED BELOW*/)
    WRITE(6,103)(ITEM(I),DES(1,I),DES(2,I),SHAPE(I),RHO(I),A(I),-F(I),
    IC(I),D(I),F(I),      XI(I),YI(I),ZI(I),XJ(I),YJ(I),ZJ(I),XK(I),
    YK(I),ZK(I),I=1,IMAX)
103 FORMAT (2X*ITEM      DESCRIPTION      SHAPE      RHO
1      A      R      C
2      */15,4X,2A9,15,1X,F16.8,5F16.3//
3      ZI      XI      YJ
4      YK      ZK*/9H14.3//)
    WRITE(6,261)
261 FORMAT(3X9HCOMPONENT2X4HDATA2X6HLISTED2X5HBELOW*/)
    WRITE(6,250)
250 FORMAT(1H0* ITEM      DESCRIPTION      WT      IXXCO
1 IYYCO      IZZCO      XCGCO      YCGCO      ZCGCO      IXYCO      IYZCO
2 IYZCO*/)
    TX=0.
    XMOM=0.
    YMOM=0.
    ZMOM=0.

```

```

      IXX=0.
      IYY=0.
      IZZ=0.
      IXY=0.
      IXZ=0.
      IYZ=0.
      IXX0=0.
      IYY0=0.
      IZZ0=0.
      PI=3.141593
      CONS=4632.
      DO50 I=1,IMAX
      XL(I)=0.
      YL(I)=0.
      ZL(I)=0.
      IXYCG(I)=0.
      IXZCG(I)=0.
      IYZCG(I)=0.
      LGTH=SQRT((XJ(I)-XI(I))**2+(YJ(I)-YI(I))**2+(ZJ(I)-ZI(I))**2)
      ITEMP=SHAPE(I)
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13),ITEMP

C      DISCRETE MASS      *SHAPE 1*
C      POINT J MUST BE SELECTED ANYWHERE ON THE X AXIS FOR DIR, COSINES

1      A(I)=RHO(I)
      IXXCG(I)=A(I)*CONS
      IYYCG(I)=B(I)*CONS
      IZZCG(I)=C(I)*CONS
      IF((XJ(I)-XI(I)).NE.0.).OR.(ABS(YJ(I)).NE.0.).OR.(ABS(ZJ(I)).NE.0,
1)) GO TO 704
      LGTH=10.
      XJ(I)=1.1*XI(I)+10.
      IF(XJ(I).EQ.XI(I))XJ(I)=XI(I)+10.
704    XL(I)=0.0
      GO TO 60

C      CYLINDER      *SHAPE 2*

2      IF(B(I).EQ.A(I))GO TO 33
      IF(R(I).LT.0.)R(I)=0.
      IF(C(I).LT.0.)C(I)=0.
      VOLCYL=PI*(A(I)**2-R(I)**2)*(LGTH-C(I))
      A(I)=RHO(I)*VOLCYL
      IF(RHO(I).GT.0.4) A(I)=RHO(I)
      RHO(I)=A(I)/VOLCYL
      IXXCG(I)=.5*A(I)*(A(I)**2+R(I)**2)
      IYYCG(I)=.25*A(I)*(A(I)**2+8(I)**2)+A(I)*(LGTH**3-C(I)**3)
      S/(12.*(LGTH-C(I)))
      IZZCG(I)=IYYCG(I)
      XL(I)=LGTH/2.
      GO TO 60

```

CYLINDER (THIN WALL) *SHAPE 2*

```

33 * (I)=RHO(I)
    XL(I)=LGTH/2.
    IXXCG(I)=(RHO(I)*A(I)**2)
    IYYCG(I)=.5*RHO(I)*A(I)**2+RHO(I)*(LGTH**3-C(I)**3)/
2((LGTH-C(I))*12.0)
    IZZCG(I)=IYYCG(I)
    GO TO 60

```

TRUNCATED CONE *SHAPE 3* (REF=-R. HULL)

```

3 IF(B(I).EQ.A(I))GO TO 30
  VOL =
51.0472*LGTH*(A(I)**2+A(I)*C(I)+C(I)**2-B(I)**2-B(I)*D(I)+D(I)**2)
  * (I) = RHO(I)*VOL
  IF(RHO(I).GT.0.4) * (I)=RHO(I)
  RHO(I) = * (I)/VOL
  XL(I)= (LGTH**2)*(A(I)**2+2.*A(I)*C(I)+3.*C(I)**2+R(I)**2+2.*B(I)*
5D(I)-3.*D(I)**2)*.2618/VOL
  KI =.05*3.141593*RHO(I)*((C(I)**5-A(I)**5)*LGTH/(C(I)-A(I))+
5D(I)**5-B(I)**5)*LGTH/(D(I)-B(I)))
  *II=3.1416*RHO(I)*LGTH**3*((A(I)**2-B(I)**2)/3+.5*((A(I)+
5C(I))*A(I)-(B(I)+D(I))*B(I))+.2*((A(I)-C(I))**2-(B(I)-D(I))**2))
  *III=1.0472*(LGTH)*(A(I)**2+A(I)*C(I)+C(I)**2-B(I)**2-B(I)*D(I)-
5D(I)**2)*RHO(I)*XL(I)**2
  IXXCG(I)=2.0*(KI)
  IYYCG(I)=KI+*II+*III
  IZZCG(I)=IYYCG(I)
  GO TO 60

```

TRUNCATED CONE (THIN WALL) *SHAPE 3*

```

30 VOL=F(I)*SQRT((A(I)-C(I))**2+LGTH**2)*3.14159*(A(I)+C(I))
  * (I) = VOL*RHO(I)
  XL(I)=LGTH/3.*((2.*C(I)+A(I))/(C(I)+A(I)))
  IYYCG(I)=(RHO(I)/4.*(A(I)**2+C(I)**2)+RHO(I)*LGTH**2/18.*(1.+2.*A(
1I)*C(I)/(A(I)+C(I))**2))
  IZZCG(I)=IYYCG(I)
  IXXCG(I)=(RHO(I)/2.*(A(I)**2+C(I)**2))
  GO TO 60

```

TORUS *SHAPE 4*

```

4 A(I)=A(I)*LGTH
  IF(D(I).LT.0.)D(I)=0.
  VOL1=2.*PI **2*LGTH**2*A(I)
  VOL2=2.*PI **2*D(I)**2*A(I)
  ACVOL=VOL1-VOL2
  * (I)=RHO(I)*ACVOL
  YMI=(RHO(I)*VOL1)

```



```

X11=.125*XM1*(4.*A(I)**2+5.*LGTH**2)
XM2=(RHO(I)*VOL2)
X12=.125*XM2*(4.*A(I)**2+5.*D(I)**2)
IYVCG(I)=(X11-X12)
IZZCG(I)=IYVCG(I)
Y11=.25*XM1*(4.*A(I)**2+3.*LGTH**2)
Y12=.25*XM2*(4.*A(I)**2+3.*D(I)**2)
IXXCG(I)=Y11-Y12
GO TO 60

```

C BEAM (ARBITRARY CROSS SECTION) *SHAPE 5*

```

5 XL(I)=LGTH/2.
IXXCG(I)=(RHO(I)*LGTH*(B(I)+C(I)))
IYVCG(I)=(RHO(I)*(B(I)*LGTH+.0833*A(I)*LGTH**3))
IZZCG(I)=(RHO(I)*(C(I)*LGTH+.0833*A(I)*LGTH**3))
VOL=A(I)*LGTH
*(I)=RHO(I)*VOL
GO TO 60

```

C CIRCULAR ROD (ARBITRARY CROSS SECTION) *SHAPE 6*

```

6 *(I)=RHO(I)
RSCG=(A(I)**2*C(I)+B(I))/(A(I)*C(I))
IXXCG(I)=RSCG**2**(I)
IYVCG(I)=.5*RSCG**2**(I)
IZZCG(I)=IYVCG(I)
GO TO 60

```

C SPHERICAL SEGMENT *SHAPE 7*

```

7 F(I)=C(I)-R(I)
D(I)=C(I)-LGTH
G=LGTH-B(I)
VOL1=1.0472*LGTH**2*(3.*C(I)-LGTH)
VOL2=1.0472*G**2*(3.*F(I)-G)
ACVOL=VOL1-VOL2
*(I)=ACVOL*RHO(I)
IF(RHO(I).GT.0.4) *(I)=RHO(I)
RHO(I)=*(I)/ACVOL
XBAR1=.75*(2.*C(I)+LGTH)**2/(3.*C(I)-LGTH)
XBAR2=.75*(2.*F(I)+G)**2/(3.*F(I)-G)
XLT=(XBAR1*VOL1-XBAR2*VOL2)/ACVOL
XL(I)=XLT-D(I)
XM1=VOL1*RHO(I)
XM2=VOL2*RHO(I)
TM=XM1-XM2
XI=(2.*LGTH*XM1/(3.*C(I)-LGTH))*(C(I)**2-.75*C(I)*LGTH+.15*LGTH
1**2)

```

```

X12=(2, *G *XM2/(3, *F(I)=G ))*(F(I)**2=.75*F(I)*G +.15*G
1**2)
IXXCG(I)=(X11-X12)
TEMP1=.05236*RHO(I)*(15, *C(I)**4*C(I)=10, *C(I)**2*C(I)**3+3, *C(I)
1**5)+.20944*RHO(I)*(5, *C(I)**2*C(I)**3=3, *C(I)**5)
TEMP2=.05236*RHO(I)*(15, *C(I)**4*D(I)=10, *C(I)**2*D(I)**3+3, *D(I)
1**5)+.20944*RHO(I)*(5, *C(I)**2*D(I)**3=3, *D(I)**5)
TEMP3=.05236*RHO(I)*(15, *F(I)**4*F(I)=10, *F(I)**2*F(I)**3+3, *F(I)
1**5)+.20944*RHO(I)*(5, *F(I)**2*F(I)**3=3, *F(I)**5)
TEMP4=.05236*RHO(I)*(15, *F(I)**4*D(I)=10, *F(I)**2*D(I)**3+3, *D(I)
1**5)+.20944*RHO(I)*(5, *F(I)**2*D(I)**3=3, *D(I)**5)
ACTEMP=(TEMP1-TEMP2)-(TEMP3-TEMP4)=(TM*XLT **2)
IYYCG(I)=ACTEMP
IZZCG(I)=IYYCG(I)
GO TO 60

```

2 SPHERE *SHAPE 8*

```

8 LGTH=A(I)
XJ(I)=XI(I)+LGTH
YJ(I)=YI(I)
ZJ(I)=ZI(I)
KJ(I) = KI(I)+10.
IF(B(I).EQ.A(I))GO TO 34
VOL1=4.188791*A(I)**3
VOL2=4.188791*B(I)**3
ACVOL=VOL1-VOL2
A(I)=RHO(I)*ACVOL
XM1=(RHO(I)*VOL1)
XM2=(RHO(I)*VOL2)
X11=(.4*XM1*LGTH**2)
Y12=(.4*XM2*B(I)**2)
IXXCG(I)=(X11-Y12)
IYYCG(I)=IXXCG(I)
IZZCG(I)=IXXCG(I)
GO TO 60

```

3 SPHERE (THIN *ALL) *SHAPE 8*

```

34 A(I)=RHO(I)
IXXCG(I)=(.667*RHO(I)*A(I)**2)
IYYCG(I)=IXXCG(I)
IZZCG(I)=IXXCG(I)
XL(I)=0.0
GO TO 60

```

4 SEMISPHERE *SHAPE 9*

```

9 IF(LGTH.EQ.A(I))GO TO 35
IF(A(I).LT.0.)A(I)=0.
VOL1=2.09439*LGTH**3
VOL2=2.09439*A(I)**3

```

```

ACVOL=VOL1-VOL2
W(I)=RHO(I)*ACVOL
XBAR1=.375*LGTH
XBAR2=.375*A(I)
XL(I)=(XBAR1*VOL1-XBAR2*VOL2)/ACVOL
XM1=(RHO(I)*VOL1)
XM2=(RHO(I)*VOL2)
XI1=(.4*XM1*LGTH**2)
XI2=(.4*XM2*A(I)**2)
IXXCG(I)=(XI1-XI2)
XI3=(.26*XM1*LGTH**2)
XI4=(.26*XM2*A(I)**2)
IYYCG(I)=(XI3-XI4)
IZZCG(I)=IYYCG(I)
GO TO 60

```

C HEMISPHERE (THIN WALL) *SHAPE 9*

```

35 A(I)=RHO(I)
XL(I)=LGTH/2.
IXXCG(I)=.666*RHO(I)*LGTH**2
IYYCG(I)=.4166*RHO(I)*LGTH**2
IZZCG(I)=IYYCG(I)
GO TO 60

```

C PARALLELEPIPED *SHAPE 10*

```

10 IF(D(I).EQ.A(I))GO TO 36
IF(D(I).LT.0.)D(I)=0.
VOL1=LGTH*B(I)*A(I)
VOL2=C(I)*F(I)*D(I)
ACVOL=VOL1-VOL2
A(I)=RHO(I)*ACVOL
IF(RHO(I).GT.0.4) A(I)=RHO(I)
RHO(I) = A(I)/ACVOL
XL(I)=LGTH/2.
XM1=VOL1*RHO(I)
XM2=VOL2*RHO(I)
XI1=(.083333*XM1*(R(I)**2+A(I)**2))
XI2=(.083333*XM2*(F(I)**2+D(I)**2))
IXXCG(I)=(XI1-XI2)
XI3=(.083333*XM1*(LGTH**2+A(I)**2))
XI4=(.083333*XM2*(C(I)**2+D(I)**2))
IYYCG(I)=(XI3-XI4)
XI5=(.083333*XM1*(LGTH**2+R(I)**2))
XI6=(.083333*XM2*(C(I)**2+F(I)**2))
IZZCG(I)=(XI5-XI6)
GO TO 60

```

PARALLELEPIPED (THIN WALL) *SHAPE 10*

```

36 XL(I)=LGTH/2.
   W(I)=RHO(I)
   TEMP1=(LGTH*B(I)*A(I))
   TEMP2=(LGTH*B(I)+B(I)*A(I)+LGTH*A(I))
   IXXCG(I)=(.083333*RHO(I)*(B(I)**2+A(I)**2)+(RHO(I)/6.)*(TEMP1*
1(B(I)+A(I))/TEMP2))
   IYYCG(I)=(.083333*RHO(I)*(LGTH**2+A(I)**2)+(RHO(I)/6.)*(TEMP1*
1(LGTH+A(I))/TEMP2))
   IZZCG(I)=(.083333*RHO(I)*(LGTH**2+B(I)**2)+(RHO(I)/6.)*(TEMP1*
1(LGTH+B(I))/TEMP2))
GO TO 60

```

SWEPT TRAPEZOIDAL PANEL (THICK WALL) *SHAPE 11* (REF=O, HULL)

```

11 W(I)=A(I)*RHO(I)*(LGTH*(H(I)+F(I))/2.)
   IF(RHO(I).GT.0.4) W(I)=RHO(I)
   RHO(I)=(W(I))/((B(I)+F(I))/2.0)*LGTH*A(I)
   XL(I)=LGTH*(B(I)+2.0*F(I))/(3.0*(B(I)+F(I)))
   XT = F(I)*LGTH/(B(I)-F(I))
   TANTAN = C(I)/LGTH
   FETAN = (F(I)/2.-B(I)/2.+C(I))/LGTH
   AFTAN = (C(I)-F(I)/2.0+B(I)/2.0)/LGTH
   THRB=LGTH+XT
   RCG=((FETAN+AFTAN)/2.)*(THRB-XL(I))
   XPAN=AFTAN-FETAN
   IYYCG(I)= W(I)*((XPAN**3*(THRB**2-XT**2))/24.+ XPAN*A(I)*
2(THRB**4-XT**4)/4.)*((THRB-XL(I))**2)**(I)
   XFRXX= XPAN*A(I)*(RCG**2*(THRB**2-XT**2)/2.-RCG*(FETAN+AFTAN)*
1(THRB**3-XT**3)/3.+(FETAN+AFTAN)**2*(THRB**4-XT**4)/16.)
   IXXCG(I)= RHO(I)*(XPAN**3*A(I)*(THRB**4-XT**4)/48.+XPAN*A(I)**3
1*(THRB**2-XT**2)/24.+XFRXX)
   IZZCG(I)=A(I)*RHO(I)*(XPAN**3*(THRB**4-XT**4)/48.+XPAN*((THRB-XL(
3I))**2+RCG**2)*(THRB**2-XT**2)/2.+(2.*(THRB-XL(I))+(FETAN+AFTAN)*R
4CG)*(THRB**3-XT**3)/3.+(4.+(FETAN+AFTAN)**2)*(THRB**4-XT**4)/16.))
   IF (ABS(C(I)).LT. .001) GO TO 60
   APXCG=H(I)/2.+ABS(TANTAN*XL(I))
   H=APXCG+SQRT((X(I)-XT(I))**2+(Y(I)-YT(I))**2+(Z(I)-ZI(I))**2)
   CGX=H-APXCG
   PRB=THRB-LGTH
   SYH=H+PRB/THRB
   PRXYK=.5*F(I)*(2.*SYH-F(I)) *(LGTH+PRB/3.+PRB**2/12.)
   IXYK=THRB**2*(H**2-(H-R(I))**2)/24.-PRXYK

```

```

IXYCGP = ABS(IXYK*A(I)*RHO(I) -*(I)*CGK*XL(I))*C(I)/ABS(C(I))
PAXRC=.7854*ABS(C(I))/C(I)
IF(ABS(IXXCG(I)-IYVCG(I)).LT..001) GO TO 20
PAXRC= ATAN((2.0*IXYCGP)/(IYVCG(I)-IXXCG(I)))/2.0
IF(IXXCG(I).GT.IYVCG(I)) PAXRC =.5*(ABS(2.*PAXRC)-3.1416)*
1 PAXRC/ABS(PAXRC)
20 T*PIXX =IXXCG(I)*COS(PAXRC)**2+IYVCG(I)*SIN(PAXRC)**2+
12.0*IXYCGP*SIN(PAXRC)*COS(PAXRC)
IYVCG(I)= IXXCG(I)*SIN(PAXRC)**2+ IYVCG(I)*COS(PAXRC)**2 +
2(2.*IXYCGP*SIN(PAXRC)*COS(PAXRC) )
IXXCG(I)=T*PIXX
21 SHFTI=ABS(TAN(PAXRC))*XL(I)
XJ(I)= (XI(I)*(LGTH-XL(I))+XJ(I)*XL(I))/LGTH
YJ(I)= (YI(I)*(LGTH-XL(I))+YJ(I)*XL(I))/LGTH
ZJ(I)= (ZI(I)*(LGTH-XL(I))+ZJ(I)*XL(I))/LGTH
XI(I)=(XI(I)*(SHFTI+CGK) -YK(I)*SHFTI)/CGK
YI(I)=(YI(I)*(SHFTI+CGK) -XK(I)*SHFTI)/CGK
ZI(I)=(ZI(I)*(SHFTI+CGK) -ZK(I)*SHFTI)/CGK
LGTH=SQRT((XJ(I)-XI(I))**2+(YJ(I)-YI(I))**2+(ZJ(I)-ZI(I))**2)
XL(I) = LGTH
GO TO 60

```

C SYMMETRIC TRAPEZOIDAL PANELS (THICK WALL)*SHAPE 12* (REF--R, NULL)

```

12 LGTH = (LGTH = 2.0* D(I))/2.0
VOL=(LGTH*(R(I)+F(I)) )*A(I)
A(I)= RHO(I)*VOL
IF(RHO(I).GT.0.4) A(I)=RHO(I)
RHO(I)=(A(I)) /((R(I)+F(I)) *LGTH*A (I))
XL(I)=LGTH*(R(I)+2.0*F(I))/(3.0*(R(I)+F(I)))
IYVCG(I) = (LGTH**3)* (F(I)**2+4.*F(I)*R(I)+R(I)**2)*A(I)*RHO(I)/
2((3.*F(I)+R(I))*5)+(A(I)**2/12.+(D(I)+XL(I))**2)*A(I)
XT = F(I)*LGTH/(R(I)-F(I))
IF(R(I).LT.F(I)) XT=R(I)*LGTH/(R(I)-F(I))
THTAU = C(I)/LGTH
FETAN = (F(I)/2.+ R(I)/2.+ C(I))/LGTH
AFTAN = (C(I)-F(I)/2.0+R(I)/2.0)/LGTH
IF(R(I).LT.F(I)) LGTH = 0.0
K1=ABS(RHO(I)*A(I)*(FETAN**3-AFTAN**3)/3.0)
K1I = (R(I)*A(I)*(RHO(I))/(LGTH+XT))*(THTAU*(LGTH+XT-XL(I))**2)
K1I=ABS(K1I)
IF(R(I).LT.F(I)) LGTH = (1.0-F(I)/R(I))*XT
XTLGTH = LGTH + ABS(XT)
IXXCG(I)=(K1*((XTLGTH )**4-(XT)**4)/2.0)-K1I*((XTLGTH )**2-XT**2)
1+ A(I)**2*A(I)/12.
IYVCG(I) = IXXCG(I) + IYVCG(I)-V(I)**2*5.0
XL(I) = LGTH + D(I)
LGTH=SQRT((XJ(I)-XI(I))**2+(YJ(I)-YI(I))**2+(ZJ(I)-ZI(I))**2)
GO TO 60

```

CURVED THIN WALL PANEL *SHAPE 13*

```

13 W(I)=2.*LGTH*A(I)*B(I)*C(I)*RHO(I)
   IF(RHO(I).GT.0.4) W(I)=RHO(I)
   RHO(I)=W(I)/(2.*LGTH*A(I)*B(I)*C(I))
   KI=2.*C(I)*A(I)*B(I)*LGTH*RHO(I)
   KII=LGTH**2/12.
   XL(I)=LGTH/2.
   IXXCG(I)=LGTH
   S      *RHO(I)*A(I)**3*R(I)*(2.*C(I)+(2.*SIN(C(I))**2)/C(I))
   IYYCG(I)=KI
   S      *(A(I)**2*(C(I)-SIN(C(I))*COS(C(I)))/(2.*C(I))+KII)
   IZZCG(I)=KI
   S      *(A(I)**2*(C(I)+SIN(C(I))*COS(C(I))-2.*SIN(C(I))**2/
   IC(I))/(2.*C(I))+KII)
   GO TO 60

BEGIN DIRCOS

60 IF((ABS(XK(I)).NE.0.).OR.(ABS(YK(I)).NE.0.).OR.(ABS(ZK(I)).NE.0.
   I)) GO TO 90
   XK(I)=XI(I)-(YJ(I)-YI(I))
   YK(I)=YI(I)+(XJ(I)-XI(I))
   ZK(I)=ZI(I)
   IF((YJ(I).NE.YI(I)).OR.(XJ(I).NE.XI(I)))GO TO 90
   VK(I)=LGTH*YI(I)
90 LX=(XJ(I)-XI(I))/LGTH
   MX=(YJ(I)-YI(I))/LGTH
   NX=(ZJ(I)-ZI(I))/LGTH
   T1=XK(I)-XI(I)
   T2=YK(I)-YI(I)
   T3=ZK(I)-ZI(I)
   LZ=MX*T3-T2*NX
   MZ=NX*T1-T3*LX
   NZ=T2*LX-T1*MX
   T4=SQRT(LZ**2+MZ**2+NZ**2)
   LZ=LZ/T4 MZ=MZ/T4 NZ=NZ/T4
   LY=MX*NX-MX*NZ
   MY=NZ*LX-NX*LZ
   NY=MX*LZ-LX*MZ
END DIRCOS

```

C ROTATE COMPONENT MOMENT OF INERTIA TO SYSTEM COORDINATES

```

IXXCO(I)=(IXXCG(I)*(LX)**2+IYYCG(I)*(LY)**2+IZZCG(I)*(LZ)**2)/CONS
IYYCO(I)=(IXXCG(I)*(MX)**2+IYYCG(I)*(MY)**2+IZZCG(I)*(MZ)**2)/CONS
IZZCO(I)=(IXXCG(I)*(NX)**2+IYYCG(I)*(NY)**2+IZZCG(I)*(NZ)**2)/CONS
IXYCO(I)=((IXXCG(I)*(LX)*(MX)+IYYCG(I)*(LY)*(MY)+IZZCG(I)*(LZ)*(M
IZ)))/CONS
IXZCO(I)=((IXXCG(I)*(LX)*(NX)+IYYCG(I)*(LY)*(NY)+IZZCG(I)*(LZ)*(N
IZ)))/CONS
IYZCO(I)=((IXXCG(I)*(MX)*(NX)+IYYCG(I)*(MY)*(NY)+IZZCG(I)*(MZ)*(N
IZ)))/CONS

```

C CALCULATE COMPONENT CENTER OF MASS COORDINATES AND WRITE OUT

```

XCG(I)=XI(I)+XL(I)*LX
YCG(I)=YI(I)+XL(I)*MX
ZCG(I)=ZI(I)+XL(I)*NX
WRITE(6,300)ITEM(I),DES(1,I),DES(2,I),*(I),IXXCO(I),IYYCO(I),IZZCO
1(I),XCG(I),YCG(I),ZCG(I),IXYCO(I),IYZCO(I),IXZCO(I)
300 FORMAT(15,5X,2A9,6F11.5,4F9.5/)

```

C CALCULATE SYSTEM WEIGHT, SECOND MOMENT AT ORIGIN AND C.G.

```

T=ST+*(I)
IXXO=IXXO+IXXCO(I)+*(I)*(ZCG(I)**2+YCG(I)**2)/CONS
IYYO=IYYO+IYYCO(I)+*(I)*(ZCG(I)**2+XCG(I)**2)/CONS
IZZO=IZZO+IZZCO(I)+*(I)*(XCG(I)**2+YCG(I)**2)/CONS
XMOM=XMOM+*(I)*XCG(I)
YMOM=YMOM+*(I)*YCG(I)
50 ZMOM=ZMOM+*(I)*ZCG(I)

```

C COMPUTE SYSTEM C.G. COORDINATES

```

XBAR=XMOM/T
YBAR=YMOM/T
ZBAR=ZMOM/T
WRITE(6,262)
262 FORMAT(1H C* SYSTEM DATA LISTED BELOW (T=LBS, INERTIAS=SLUGS FT SQ
1UARED, C.G.=INS, SECOND MOMENT=SLUG FT SQUARED)*//)
WRITE(6,100)T,IXXO,IYYO,IZZO,XBAR,YBAR,ZBAR
100 FORMAT(11X3HSYS1X2H T12X4H IXX014X4H IYY014X4H IZZ016X4H XBAR14X4H YBAR
114X4H ZBAR/7F18.3)

```

ORIGINAL PAGE IS
OF POOR QUALITY

TRANSFER MASS PROPERTIES TO SYSTEM C.G., SUM AND WRITE OUT

```

DO70 I=1,IMAX
DELX=XCG(I)-XBAR
DELY=YCG(I)-YBAR
DELZ=ZCG(I)-ZBAR
IXX=IXX+IXXCO(I)+W(I)*(DELY**2+DELZ**2) /CONS
IYY=IYY+IYYCO(I)+W(I)*(DELX**2+DELZ**2) /CONS
IZZ= IZZ+IZZCO(I)+W(I)*(DELY**2+DELX**2)/CONS
IXY=IXY+IXYCO(I)+W(I)*DELX*DELY/CONS
IXZ=IXZ+IXZCO(I)+W(I)*DELX*DELZ/CONS
70 IYZ=IYZ+IYZCO(I)+W(I)*DELY*DELZ/CONS
+RITE(6,200)IXX,IYY,IZZ,IXY,IXZ,IYZ
200 FORMAT(12X3HIXX16X3HIYY15X3HIZZ15X3HIXY15X3HIXZ15X3HIYZ/6F18.5/)

```

COMPUTE INERTIAS (EIGENVALUES) ABOUT PRINCIPAL AXES AND EACH AXIS
DIRECTION COSINES (EIGENVECTORS) AND WRITE OUT

```

PRINT 340
340 FORMAT(/1X,*INERTIAS (EIGENVALUES) ABOUT SYSTEM PRINCIPAL AXES WIT
1H AXIS DIRECTION COSINES (EIGENVECTORS) RELATING THE PRINCIPAL AXE
1S*/1X,*TO THE X, Y, AND Z SYSTEM AXES IN THAT SEQUENCE*///)
MAX=3
N=3
ARR(1,1)=IXX
ARR(2,1)=ARR(1,2)=IXY
ARR(1,3)=ARR(3,1)=IXZ
ARR(2,2)=IYY
ARR(3,2)=ARR(2,3)=IYZ
ARR(3,3)=IZZ
CALL SYMDI(MAX,N,ARR,E,CRR,IERR)
IF(IERR.NE.0) GO TO 332
DO 336 J=1,3
PRINT 337,J,E(J)
337 FORMAT(1X,*EIGENVALUE(*I1*) = *F12.5//)
PRINT 339,J
339 FORMAT(1X,*EIGENVECTOR(*I1*)*)
PRINT 338,(ARR(I,J),I=1,3)
338 FORMAT(1X,3(F14.6,5X)///)
336 CONTINUE
GO TO 334
332 PRINT 333,IERR
333 FORMAT(1X,*ERROR == IERR = *I5)
334 STOP
END

```


1. Report No. NASA TM 78681		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle Computer Program for Determining Mass Properties of a Rigid Structure				5. Report Date March 1978	
				6. Performing Organization Code	
7. Author(s) Reid A. Hull, Dr. John L. Gilbert and Phillip J. Klich				8. Performing Organization Report No.	
				10. Work Unit No.	
9. Performing Organization Name and Address National Aeronautics and Space Administration Langley Research Center Hampton, Virginia 23665				11. Contract or Grant No.	
				13. Type of Report and Period Covered Technical Memorandum	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546				14. Sponsoring Agency Code	
15. Supplementary Notes Tentatively will become a Reference Publication					
16. Abstract A method is presented for determining the complete mass properties of a rigid structure. Computer technology is utilized to exercise the greatest savings possible in solving for and printing out this information including properties relative to the total structure's principal axes.					
17. Key Words (Suggested by Author(s)) Mass Properties Computer Rigid Structure				18. Distribution Statement Unclassified - Unlimited	
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 46	
				22. Price* \$4.50	

